Blockchain: Research and Applications 3 (2022) 100067

Contents lists available at ScienceDirect

Blockchain: Research and Applications

journal homepage: www.journals.elsevier.com/blockchain-research-and-applications

A survey on blockchain technology and its security

Huaqun Guo [a,*], Xingjie Yu [b,1]

[a] Institute for Infocomm Research, A*STAR, Singapore
[b] Singapore

ARTICLE INFO

Keywords:
Blockchain
Consensus algorithm
Smart contract
Risk
Blockchain security

ABSTRACT

Blockchain is a technology that has desirable features of decentralization, autonomy, integrity, immutability, verification, fault-tolerance, anonymity, auditability, and transparency. In this paper, we first carry out a deeper survey about blockchain technology, especially its history, consensus algorithms' quantitative comparisons, details of cryptography in terms of public key cryptography, Zero-Knowledge Proofs, and hash functions used in the blockchain, and the comprehensive list of blockchain applications. Further, the security of blockchain itself is a focus in this paper. In particular, we assess the blockchain security from risk analysis to derive comprehensive blockchain security risk categories, analyze the real attacks and bugs against blockchain, and summarize the recently developed security measures on blockchain. Finally, the challenges and research trends are presented to achieve more scalable and securer blockchain systems for the massive deployments.

1. Introduction

In blockchain, data are kept in a distributed ledger. It is the blockchain technology to provide integrity and availability that allows participants in the blockchain network to write, read, and verify transactions recorded in a distributed ledger. However, it does not allow the deletion and modification operations on the transactions and other information stored on its ledger. The blockchain system is supported and secured by cryptographic primitives and protocols, e.g., digital signatures, hash functions, etc. These primitives guarantee the transactions that are recorded into the ledger are integrity-protected, authenticity-verified, and non-repudiated. Further, as a distributed network, to allow the entire set of participants to agree on a unified record, the blockchain technology also needs a consensus protocol, which is essentially a set of rules to be followed by every participant, in order to achieve a globally unified view.

In a trustless environment, blockchain provides users with desirable features of decentralization, autonomy, integrity, immutability, verification, fault-tolerance, attracted great academic and industrial attention in the recent few years, anonymity, auditability, and transparency [1–3]. With these advanced features, blockchain technology has attracted great academic and industrial attention in the recent few years.

To help and benefit someone to understand the blockchain technology and blockchain security issues, especially for users who use blockchain to do the transactions, and for researchers who will be developing blockchain technology and addressing blockchain security issues, we put in our effort and time to conduct the comprehensive survey and analysis on blockchain technology and its security issues. First, we identify keywords, namely, blockchain, survey, consensus algorithm, smart contract, risk, and blockchain security to search publications and information on the Internet. Second, we survey papers related to blockchain published in top security conferences and journals, e.g., USENIX Security Symposium, IEEE Symposium on Security and Privacy, IEEE Transactions journals, and so on. In this way, we have surveyed as many papers as possible so as to overcome the study and result biases. Our survey paper presents the comprehensive findings from other research work.

The main contributions of our survey include: 1) We compare various consensus algorithms with detailed analysis and numerical figures and present the cryptography fundamentals of blockchain; 2) We present the rich information about the smart contract and its security; 3) We explore the widely used applications of blockchain technology, including but not limited to different cryptocurrencies; 4) We conduct a comprehensive analysis on the security risks, real attacks, bugs, root causes, and recent security measures on blockchain itself; Last but not least, 5) The challenges and research trends are summarized and presented in this paper for the further efforts to develop the blockchain technology for the massive deployments.

* Corresponding author.
E-mail addresses: guohuaqun@u.nus.edu, guohuaqun@yahoo.com (H. Guo), stefanie_yxj@hotmail.com (X. Yu).

[1] This work has been done when the author was with Institute for Infocomm Research, A*STAR, Singapore.

https://doi.org/10.1016/j.bcra.2022.100067
Received 6 October 2021; Received in revised form 2 February 2022; Accepted 12 February 2022

The rest of the paper is organized as follows: Section 2 introduces the overview. Section 3 describes the blockchain technology in detail, including consensus algorithms, smart contracts, and cryptography for blockchain, while the comprehensive blockchain applications are presented in Section 4. The security risks and real attacks on blockchain are presented in Section 5, and security measures are described in Section 6. Section 7 analyses the challenges and the research trends for blockchain. Section 8 summarizes the related survey work to show our contribution. Finally, Section 9 concludes our work.

2. Overview of blockchain history

In 1982, Chaum was the first known person to propose a blockchain-like protocol in his Ph.D. thesis [4]. In 1991, Haber and Stornetta described a secured chain of blocks cryptographically [5]. In 1993, Bayer et al. incorporated Merkle trees into the design [6]. In 1998, "bit gold''—a decentralized digital currency mechanism was designed by Szabo [7]. In 2008, Nakamoto introduced Bitcoin, electronic cash with a purely peer-to-peer network [8]. It was also in 2008 that the term blockchain was first introduced as the distributed ledger behind Bitcoin transactions [9].

In 2013, Buterin proposed Ethereum in his whitepaper [10]. In 2014, the development of Ethereum was crowdfunded, and on July 30, 2015, the Ethereum network went live. The emerging of Ethereum implied that blockchain 2.0 was born because different from all the various blockchain projects that focused on developing altcoins (other coins which are similar to Bitcoin), Ethereum enables people to connect through trustless distributed applications on its own blockchain. In other words, while Bitcoin is developed for distributed ledger, Ethereum is developed for a distributed data storage plus smart contracts, which are small computer programs. Ethereum 2.0 upgrades the Ethereum network which aims to boost the speed, scalability, efficiency, and security of the network. The upgrades have 3 phases crossing from 2020 to 2022.

In 2015, the Linux Foundation announced the Hyperledger project, which is open-source software for blockchains. With the aim of building enterprise blockchain, Hyperledger blockchain frameworks are different from Bitcoin and Ethereum. Under Hyperledger, there are eight blockchain frameworks, including Hyperledger Besu, Hyperledger Fabric, Hyperledger Indy, Hyperledger Sawtooth, Hyperledger Burrow, Hyperledger Iroha, Hyperledger Grid, and Hyperledger Labs, five Hyperledger tools, including Hyperledger Avalon, Hyperledger Cactus, Heperledger Caliper, Hyperledger Cello, and Hyperledger Explorer, and four libraries, including Hyperledger Aries, Hyperledger Quilt, Hyperledger Transact, and Hyperledger URSA [11].

The history of blockchain is summarized in Fig. 1. Bitcoin and Ethereum are public blockchains since anyone can participate in their blockchain networks, which are also called permissionless blockchains. The various Hyperledger blockchain networks are private blockchains since the participants are needed to be verified first before joining the network, which are also called permissioned blockchains.

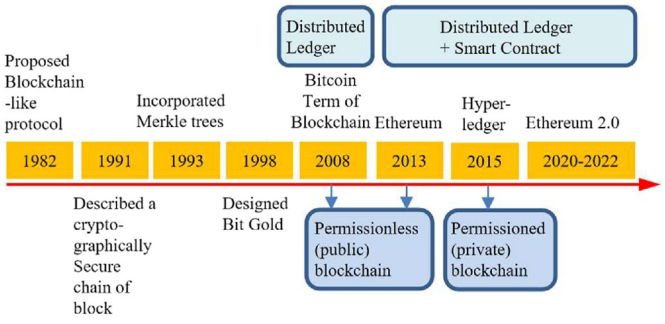

Fig. 1. History of blockchain.

3. Blockchain Technology

3.1. Consensus algorithms

As one of the desired blockchain features, anonymity also poses a problem when it comes to trust. How can it be 100% ensured that anonymous users are honest when they add transactions to a ledger? The answer is to validate every transaction to be legal (not malicious, double-spending, etc.) and then put the transactions into a block. The agreement of adding a block to the blockchain is through consensus algorithms. These consensus algorithms take advantage of the fact that the majority of users on a blockchain have a common interest in keeping the blockchain honest. A blockchain system uses a consensus algorithm to build its trust and properly stores the transactions on the blocks. Thus, consensus algorithms can be considered the heart of all transactions of blockchains.

A consensus protocol is essentially a set of rules to be followed by every participant. As a distributed technology without a universal trust, blockchain needs a distributed consensus mechanism for all participants to agree on the blockchain's current state. The blockchain's consensus is based on scarcity that controlling more of a scarce resource gives more control over the blockchain's operation. A number of unique consensus mechanisms have been designed for blockchains, which include Proof of Work (PoW) [8], Prof of State (PoS) [12], Delegated Proof of State (DPoS) [13], Proof of Elapsed Time (PoET) [14], Practical Byzantine Fault Tolerance (PBFT) [15], Directed Acyclic Graph (DAG) [16,17], Proof of Authority (PoA) [18], Tendermint [19], Ripple [20], Scalable Byzantine Consensus Protocol (SCP) [21], Proof of Bandwidth (PoB) [22], Proof-of-Importance (PoI) [23], Proof of Burn [24], Proof of Capacity [25], depending on their unique requirements.

Based on algorithms that appeared in other survey work [2,3,26–32], PoW, PoS, DPoS, and PBFT are the most common consensus algorithms. DAG is the most different from other consensus algorithms. PoET is developed by Intel Corporation and used in Hyperledger Sawtooth. Thus, these six consensus algorithms are further described below.

Proof of Work (PoW). PoW selects a problem that can only be solved by guessing. For example, when it is time to create and validate a full block, the problem is to guess a nonce value such that when using the transaction data and the nonce value as inputs for a hash function, its hash output needs to match the difficulty, e.g., beginning with four leading zeros. Every node (also called mining node) on the network is now guessing different nonce values randomly until one node first happens to find the nonce value that matches the difficulty. So a mining node has to spend a lot of computational resources on it (hence called "work") and solves the problem faster than others in order to succeed in creating a block to link to the blockchain, and obtain an incentive mining reward, which is often cryptocurrency. On the other hand, hash functions are important as one cryptographic puzzle at the center of the PoW consensus algorithm. The Bitcoin network adopts the cryptographic hash function SHA-256 [8]. We will talk more about the hash function in the following section. Bitcoin and Ethereum public blockchains use PoW as their consensus algorithm. A big issue with the PoW consensus process is that it requires a lot of time and electricity to complete.

Proof of Stake (PoS). PoS [12,33] is the second most prominent consensus method and requires fewer computations for mining than PoW. PoS solves time and electricity consumption problems that PoW has because the electricity requirement is associated with miners finding a nonce and this process needs to take some time. PoS has nodes to put up a stake to be chosen as the next block creator. When a block is chosen, the creator will receive the transaction fees associated with that block. If a block winner attempts to add an invalid block, he/she will lose his/her stake. In its first phase of the Ethereum 2.0 upgrade, the blockchain "world computers" switch from PoW to PoS consensus algorithm.

Delegated Proof-of-Stake (DPoS). In DPoS, all token holders can vote for a number of delegates and can also delegate to other users with their voting power. The more tokens that the token holder has, the more

voting power the token holder has. Then the delegates are responsible for validating transactions and blocks to secure the network [13]. Unlike the most computing power in PoW or the most tokens in PoS, token holders in DPoS are allowed to vote on who to mine new blocks and reward only the best miners. EOS is one of the blockchain systems to use the DPoS algorithm [34].

Proof of Elapsed Time (PoET). Intel Corporation developed PoET to enable a different way to determine a winner to mine a block [14]. In PoET, each potential validation node requests a random waiting time which is generated on a trusted computing platform, e.g., Intel's SGX. After waiting for the assigned time, the first node that finished waiting time is the validation winner and is able to add the new block. The trusted computing platform enables every node to have a chance to be the winner [35].

Practical Byzantine Fault Tolerance (PBFT). Byzantine Fault Tolerance (BFT) is to solve a famous general problem that some generals are dishonest but needs to reach a correct consensus. PBFT is a consensus algorithm that optimizes BFT [16]. In PBFT, as long as the malicious or hostile nodes are less than one third of all the nodes in the blockchain system, the blockchain system will come to agree on the blockchain's current state. The more nodes in the blockchain system, the securer the blockchain is. Hyperledger Fabric currently uses PBFT.

Directed Acyclic Graph (DAG). DAGs [17] are made up of vertices and the edges (the lines connecting them), which is different from other consensus algorithms. The vertices and the edges are directed because they head in one direction and they are acyclic because the vertices do not loop back on themselves. Each vertex in the structure represents a transaction. There is no notion of blocks here, and mining is not required to add transactions. Instead of gathering transactions into blocks, each transaction is built on top of another. Still, there is a small PoW operation that is done when a node submits a transaction. This ensures that the network is not being spammed and also validates previous transactions. IOTA [36] adopts the DAG consensus algorithm.

The comparisons of these six consensus algorithms are listed in Table 1. We compare them in as many details and as much quantitative as possible.

3.2. Smart contract

The smart contract makes another beautiful part of blockchain that blockchain not only provides a distributed, unchangeable record of all the different events that have occurred, but also allows to write very non-subjective computer code that defines exactly how that process is going to be managed and what steps are going to be taken when that event occurs. One goal of the smart contract proposed in Ethereum was to break the limitations of Bitcoin. The smart contract is about computer code that is written to respond to certain types of significant events. The smart contract does not have to involve two or more parties and does not have to be legally binding [41].

Table 1
Comparison of consensus algorithms [13,37–40].

	PoW	PoS	DPoS	PoET	PBFT	DAG
Setup	Public permissionless/ Private blockchain	Public permissionless/ Private blockchain	Public/Private blockchain	Private permissioned/ permissionless blockchain	Private permissioned blockchain	Public permissioned non-blockchain
Cost of entry and returns	Relatively high cost of entry, but high returns	Low cost of entry, but low returns	Lower cost and lower returns than PoS	Very low cost of entry, but low returns	All participate with no return	All participate with no return
Incentives	The winning miner receives new coins with the block & transaction fees in the block he/she validates	The winner receives transaction fees with the new block. If a block winner attempts to add an invalid block, he/she loses his/her stake	The threat of loss of reputation & income provides an incentive for delegates to act honestly and keep the network secure	The winning miner receives the transaction fees with the new block he/ she validates.	Nil	Nil
Finality	Probabilistic	Probabilistic	Probabilistic	Probabilistic	Immediate	Probabilistic
Scalability in network	High	Medium	Medium	Medium	Low (quickly grow into a huge communication cost as the amount of nodes scales upwards)	High
Energy efficiency	Very low (energy intensive computations, e.g., Bitcoin consumes around 121.36 terawatt-hours (TWh) a year)	High	High (no miners required)	High	Medium (Some PBFT systems use PoW to prevent Sybil attack, but only after a set number of blocks (i.e., 100) and not for every block)	Medium (A small PoW operation when a node submits a transaction to ensure the network is not being spammed and also validates previous transactions)
Majority or 51% attack	The number of malicious nodes >25% of all nodes for attack	Reduced 51% attack probability	Easier to organize a 51% attack if delegates combine their power	Reduced 51% attack probability	The number of malicious nodes > one third of all nodes for attack	Not tested at scale
Susceptible to Sybil attack	No	Yes	Yes	No	Yes	No
Examples	Bitcoin, Ethereum, Litecoin, Monero, Dash, Zcash, Decred, and more	Ethereum 2.0, Cardano, Polkadot BlackCoin, and Peercoin.	EOS, BitShares, Lisk, Steem, Ark, Nano, Cardano, and Tezos.	Hyperledger Sawtooth	Hyperledger Fabric, Zilliqa	IOTA
Transactions per second (TPS)	Bitcoin: 7 maximum 27	Ethereum: 15	EOS: 3996 BitShares: 3300	Hyperledger Sawtooth: 2300	Hyperledger Fabric: approximately 3500	IOTA: 250 IOTA Pollen V0.2.2: >1000
Block confirmation time (s)	Bitcoin: 6000 Litecoin: 150	Ethereum: 15	EOS: 0.5 BitShares: 3	No actual time is found	In seconds' level (No actual time is found)	120

Smart Contract also known as chaincode [41]:

- Program rules and decision points into blockchain transactions and processes.
- Automate transactions and ensure they are all following the same rules.
- Run on the blockchain.

The smart contract will revolutionize how we do business and is the keystone for the enterprise blockchain applications. Anyone can develop smart contracts without the need for intermediaries. The smart contract provides autonomy, efficiency, accuracy, and cost-saving.

3.3. Cryptography for blockchain

Blockchain creates a layer of trust between untrusted parties to enable secure and trusted records and transactions to occur. Without blockchain to create trusted records and transactions, a third-party intermediary is necessary. blockchain uses cryptography and collaboration to create that trust and as a result, it eliminates the need for a centralized institution to act as an intermediary. Information on the blockchain is stored on the ledger using cryptography.

Blockchain makes use of some cryptography building blocks as below [41]:

- Public Key Cryptography: Be used for digital signatures and encryption.
- Zero-Knowledge Proof: Demonstrate the knowledge of a secret without revealing it.
- Hash Functions: One-way pseudo-random mathematical functions. Merkle trees adopted the hash function to form one component of the block header.

Public key cryptography. It is used to prove that a transaction was created by the right person. In blockchain, the private key is kept in a digital wallet, either a hardware wallet (a physical device to store the private key) or any software wallet (e.g., a desktop wallet app, mobile wallet app, or web-wallet). A user accesses its private key to sign a message called a digital signature that will be transmitted to the blockchain, and its public key is to confirm that the message actually did come from the user. For example, in Fig. 2, the user hashes its transaction data into hash value 1 and then signs on the hash value 1 with its private key to generate the digital signature. The user then sends its digital signature together with its transaction data to the blockchain network. The miner uses the user's public key to decrypt the received digital signature to obtain hash value A, and the miner also hashes the received transaction data to obtain another hash value B. Then the miner checks if hash value A equals hash value B or not. If they are equal, the miner verifies the user's transaction.

Since the private key is only securely kept by its owner, the corresponding digital signature makes sure the authorship of the transaction. The algorithm enables the digital signature on every transaction depending on the individual private key of each user. The pair of public key and private key fits into blockchain as a backbone of blockchain and they are used to sign and verify transactions that the user makes.

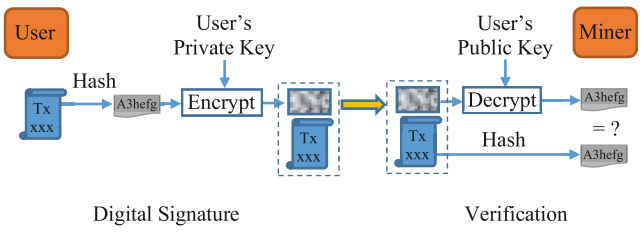

Fig. 2. Digital signature and hash used in blockchain transactions.

Both Ethereum and Hyperledger Fabric use digital signatures on transactions and blocks to confirm the identity of the creator and that the signed data has not been modified since signing. The Elliptic Curve Digital Signature Algorithm (ECDSA) is widely adopted to create a pair of public key and private key.

The public key of a user can be logically selected as an identity of the user since knowledge of a public key is necessary for the verification of digital signatures. It is used in the blockchain as a method for managing users' identities without revealing real-world identities.

Zero-Knowledge Proofs. One of the primary use cases for Zero-Knowledge Proofs in blockchain is shown in the following. When a user makes a request to send another user some money, the blockchain naturally wants to make sure, before it commits this transaction, that the user who is sending money has enough money to send. However, the blockchain does not really need to know or care who is spending the money, or how much total money he/she has. In this case, the blockchain has zero knowledge about who the user is sending the money to and how much money the user has.

Zero-knowledge proofs are a cryptographic principle used in some blockchains to increase the privacy of users. Currently, Ethereum does not have support for zero-knowledge proofs, but adding the necessary functionality for zkSNARKS, a type of zero-knowledge proof, is currently included in the Ethereum development roadmap.

Hash Functions. Hash functions are a key technology used in the blockchain. A hash function is a mathematical equation with five important properties for cryptography:

- **Fixed size.** Hash functions can take anything as an input and create an output with a fixed size. This makes it possible to condense anything into a piece of data of a fixed size. So blockchains use hash functions to condense messages for digital signatures.
- **Preimage resistance.** Given an input, it is not hard to calculate a hash output. However, given the hash output, it is mathematically impossible to reverse-engineer the original input. In fact, the only possible way is to randomly input the data into the hash function until the same output is produced.
- **2nd preimage resistance.** If an input and its hash output are given, getting the second input that produces the same hash output is computationally infeasible.
- **Collision resistance.** Finding any two distinct inputs is computationally infeasible to produce the same hash output.
- **Big change.** If any single bit of the input is changed, it will produce an entirely different hash output.

Fig. 3 shows that the cryptographic hash function provides a way to link all blocks on the blockchain together. On the block level, the hash of the previous Block i–2 header is stored in Block i–1, the hash of the previous Block i–1 header is stored in Block i, the hash of the previous Block i header is stored in Block i+1, and so on.

Within a block, there are multiple transactions. Blockchain also hashes every transaction and for a Merkle Tee at the bottom part of Fig. 3 and the Merkle Root is stored in the block header. In this way, blockchain creates a distributed ledger that is immutable, secure, and extremely trustworthy. If any block or any transaction or information on that block is modified, no matter how small, it will be discovered immediately and the link between that block and all subsequent blocks will be broken.

P2PKH address. Besides the blockchain connection structure, Merkle Tree, and the PoW mining algorithm mentioned in the previous session, cryptographic hash functions are also used in Bitcoin pay to public key hash (P2PKH) addresses [42]. Hash functions and public key cryptography are used to create the P2PKH address for the Bitcoin user to send and receive funds (Fig. 4). Due to the one-way function, it is impossible to reverse engineering from the address to its public key and private key.

The length of a key is not changed. The size of a private key is 32 bytes, and the size of a public key is 65 bytes (or 33 bytes for a compressed public key). The size of the P2PKH address is 20 bytes.

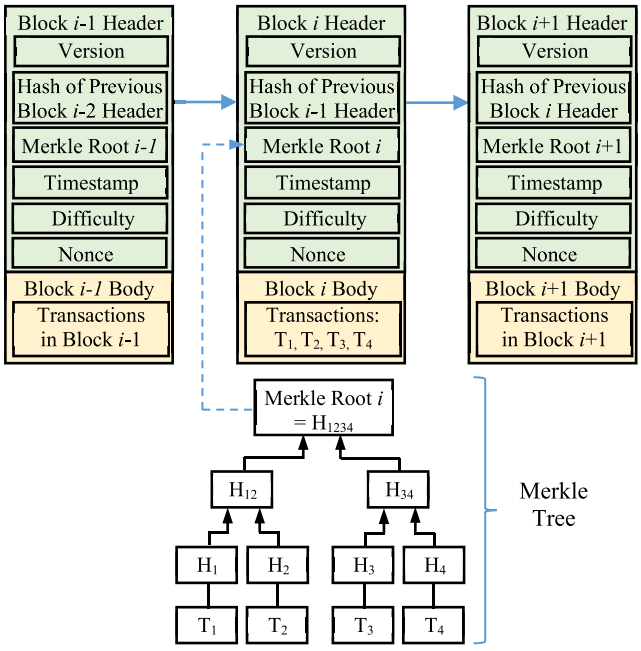

Fig. 3. Blockchain connection structure and a Merkle tree with hash function.

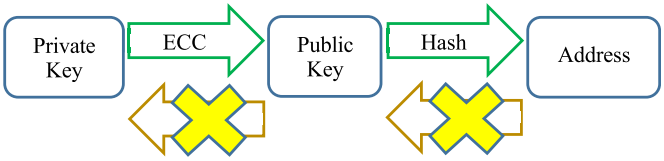

Fig. 4. Public key cryptography and hash function for Bitcoin address.

4. Blockchain Applications

From the survey, the applications of blockchain include cryptocurrency, finance (stock exchange, financial services, P2P financial market, crowdfunding, etc.), Internet-of-Things (IoT) (safety and privacy, e-business, etc.), reputation system (web community, academics, etc.), security and privacy (security enhancement, risk management, privacy protection, etc.) [3], healthcare, insurance, copyright protection, energy, society applications (blockchain music, blockchain government), advertising [43], defense, mobile applications, supply chain, automotive [28], agricultural sector [44], identity management, voting, education, law and enforcement, asset tracking [45], digital records, intrusion detection [46], digital ownership management, property title registries, and so on. Fig. 5 illustrates the spiral increasing applications of blockchain technology. It is expected that more and more use cases of blockchain systems are emerging.

In the following sub-sessions, cryptocurrency as the first application, supply chain as a widely used case, and Smart Dubai Office as the first whole government service application are selected for further information to be presented.

4.1. Cryptocurrencies

The first cryptocurrency is Bitcoin, which was announced in 2008 and launched in 2009. The maximum number of Bitcoin is 21 million BTC. Once one mining node (miner) finds a nonce value that matches the difficulty and succeeds in having a block accepted, the miner obtains a transaction fee (24 USD and 31 USD) and a mining reward of 6.25 BTC at this moment. For every 210,000 blocks (roughly every 4 years), the mining reward gets cut in half. Currently, just under 90% of BTC has been mined. After Bitcoin, the market cap of Ethereum (ETH) is roughly 19%

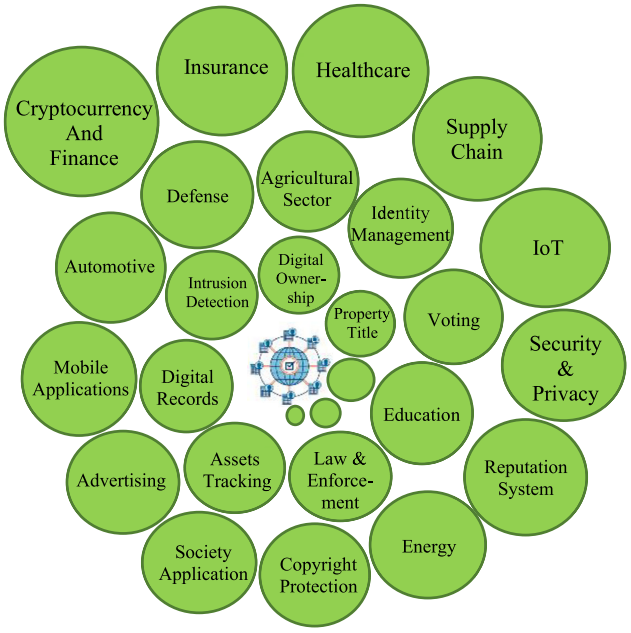

Fig. 5. Blockchain applications.

of Bitcoin's size and ranks as the second-largest cryptocurrency currently. Cryptoslate lists 2403 top cryptocurrencies by market capitalization in its coin rankings [47]. Among them, 7 cryptocurrencies that are mentioned as consensus use examples in the previous session are shown in Table 2.

Pros of cryptocurrencies include:

- Cryptocurrencies are good use cases for blockchain which make full use of the advanced features of blockchain.
- Payments go directly from one person to another.
- The processing fee is tiny.
- There are no delays and no limits for sending money.

Cons of cryptocurrencies include:

- There is no control, which may incur the black money.
- It may suffer a security attack and lose the digital assets.
- Government regulations are lacking, and some policies may be launched to manage or control cryptocurrencies.
- Some comments say that it is highly risky and speculative to invest in cryptocurrencies. For example, Tesla alerted investors about the volatility of Bitcoin's price in its SEC filing [49].

4.2. Supply chains

Blockchain technology offers distributed ledgers that create a permanent and shared record of every transaction. All recorded transactions are visible to authorized participants, traceable within the ledger, immutable and irrevocable, which prompt the increasing usage of blockchains for data sharing in supply chains. For example, IBM has released permissioned blockchain-based data sharing solutions for supply chains with a particular focus on logistics [3]; and the cold-chain logistics solution from VeChain uses blockchain to track and monitor the logistic information for transparent, regulated, secure, and reliable data sharing [4]. In Makerchain [50], twinning unique chemical signature data to blockchain is presented as an anti-counterfeiting method.

In addition, various blockchain technologies have been presented to enhance the security, transparency, and traceability of the supply chain. In Ref. [51], blockchain technology is used to secure smart manufacturing in industry 4.0 which addresses cybersecurity issues in the manufacturing systems. In Ref. [52], blockchain is used to achieve sustainability from the manufacturing system perspective and the

Table 2
Cryptocurrencies [48].

	Launched Year	Launched Price	Unit Price on Jan 1, 2021 (USD)	Unit Price on Feb 27, 2021 (USD)	Market capitalization on Feb 27, 2021 (USD)	Mined Numbers	Total Number
Bitcoin (BTC)	2009	0.0008 USD	28,994.01	47,781.33	890.6 billion	near 90%	21,000,000
Ethereum (ETH)	2014	Presale: 0.30 USD Homestead launched: 12.50 USD	737.71	1502	172.5 billion	114.84 M	Currently no implemented hard cap, & limited to 18 million per year
Cardano (ADA)	2017	0.019 EUR	0.31	1.36	43.4 billion	around 71%	45,000,000,000
Polkadot (DOT)	2020	1.2 USD	9.12	33.64	30.7 billion	1,049,328,830	Does not have a maximum supply
Litecoin (LTC)	2011	4.3 USD	124.67	176.31	11.8 billion	around 79%	84,000,000
Bitcoin Cash (BCH)	2017	543 USD	below 400	501.3	9.4 billion	near 89%	21,000,000
EOS	2017	2.29 USD	2.5975	3.68	3.5 billion	near 93%	1,027,393,754
IOTA	2016	Unknown	0.2969	1.1532	3.2 billion	2,779,530,283	2,779,530,283

product lifecycle management perspective. ManuChain [53] based on a permissioned blockchain network is presented to get rid of unbalance/inconsistency between holistic planning and local execution in individualized manufacturing systems.

4.3. Smart Dubai Office

Dubai is investing in the Smart Dubai Office and implementing blockchain technology to transform government from a service provider to a service enabler on a city-wide scale. It is funding blockchain implementation at many levels.

- Government services are implemented with blockchain technology.
- Empower startups and businesses to create the blockchain industry.
- Set up a pioneer example for the government services built upon blockchain technology.

5. Security risks and attacks with blockchain

As blockchain is decentralized without engaging any third party and needs to ensure trust in the trustless infrastructure, security on blockchain itself is worthy to conduct the research. This section will focus on security risks on blockchain technology, and a survey of real attacks and bugs on blockchain systems.

5.1. Security risks on blockchain

Top 10 Web Application Security Risks listed by the OWASP Top 10 are analyzed and assessed on blockchain technology [54], and its assessment results are summarized in Table 3. The OWASP Top 10 is a widely known document about top critical security risks in web applications, and blockchain technology faces 9 out of the top 10 risks as shown in Table 3. Thus, the security on blockchain is one of the key components to the success of blockchain business applications.

One research group surveyed and analyzed the vulnerabilities of blockchain systems from 2009 to May 2017, and listed nine categories of blockchain security risks at low level in Table 4 [29].

Another research group presented blockchain security at a higher level. They pointed out that like traditional computing, the blockchain also faces the potential attacks of Denial-of-Service (DoS), endpoint security, intentional misuse, code vulnerabilities, and data protection, but the details of launching attacks vary [41]. Other than DoS attacks, some research work also presented BGP (Border Gateway Protocol) hijacks by manipulating routing advertisements, routing attacks by delaying the propagation of blocks or isolating some parts of the blockchain network, eclipse attacks by isolating a victim from the view of the network, EREBUS attacks by making malicious transit autonomous systems (ASes) as man-in-the-middle network of Bitcoin nodes to inference the nodes' decision as a stealthier attack, DNS attacks, and remote side-channel

Table 3
Top 10 web application security risks on blockchain technology [54].

Top 10 Web Application Security Risks	Assess on blockchain Technology	Analysis Examples
Injection	Poor input sanitization in blockchain technology	Before the EOS mainnet launches, discovered vulnerability of buffer-out-of-bounds write in EOS smart contract and the potential to run the malicious smart contract
Broken Authentication	A large attack surface exists without proper implementation of authentication functionality	The cryptocurrency LISK is an example of allowing an attack on authentication
Sensitive Data Exposure	High potential for this vulnerability	Vulnerable to data mining efforts—mining the public data on blockchain for useful information; Quantum computing will break the public key cryptography used to encrypt data on the blockchain
XML External Entities (XXE)	Not applicable	
Broken Access Control	One major vulnerability for smart contracts	Two attacks on Parity multi-signature wallets due to access control vulnerabilities
Security Misconfiguration	Affect blockchain security	Attackers exploited a vulnerability to steal cryptocurrency when Ethereum wallets were configured to receive external commands from port 8545
Cross-Site Scripting (XSS)	Affect blockchain in some ways	Blockchain explorers under XSS attack could display untrusted transaction data; Both blockchain explorers and wallets under XSS attack could allow access to a private key of a user and control over his/her account
Insecure Deserialization	May compromise of blockchain systems	If malicious users control transaction data, blockchain systems may be compromised by the vulnerable deserialization code
Using Components with Known Vulnerabilities	Very common to reuse code for Ethereum smart contracts	More than 90% of smart contracts in Ethereum did reuse code, and may contain known vulnerabilities
Insufficient Logging & Monitoring	The log owners may un-monitor their logs	May smart contracts lack of monitoring and hackers may exploit their vulnerabilities without being detected

Table 4
Blockchain security risk categories at low level in Ref. [29].

S/N	Category
1	51% vulnerability
2	Criminal activity
3	Private key security
4	Transaction privacy leakage
5	Double-spending
6	Criminal smart contracts
7	Under-priced operations
8	Smart contract's vulnerabilities
9	Under-optimized smart contract

Table 5
Blockchain security risk categories at high level.

Risk	Description
Network Attacks	As shown in Table 1, blockchain has a limited number of transactions per second, DoS attacks may submit more transactions than the blockchain's capability and cause the blockchain unavailable. Besides DoS, BGP (Border Gateway Protocol) attacks, routing attacks, eclipse attacks, stealthier attacks, DNS attacks, and remote side-channel attacks are also under this category.
Endpoint Security	Endpoints can be heterogeneous which have more options to exploit the vulnerabilities. Endpoints can be also homogeneous which a flaw in one system can exist in all systems.
Intentional Misuse	As shown in Table 1, the attackers may control more nodes to launch like 51% type of attacks.
Code Vulnerabilities	Code vulnerabilities can come from smart contracts that anyone can write or the underlying platform code. The vulnerabilities have a wide-reaching impact due to the distributed network and the code cannot be modified once deployed. Intentionally write malicious smart contracts.
Data Protection	Data protection relies upon the blockchain instead of data owners to provide data integrity and availability.
Human Negligence	The log owners may un-monitor their logs.

attacks. We put those attacks under the network attacks category. Our paper adds one more risk category of human negligence since human are a weak point in any system. Table 5 lists six risk categories that may be exploited by attackers to launch attacks.

Combining Tables 3–5, we come to have the comprehensive view of security risks on blockchain shown in Table 6. Some other low level security risks such as wallet security, Sybil attacks, personal key security to highlight its importance, and liveness attack, balance attack, timejacking attack, finney attack, race attack, and SelfHolding attack which we put under the intentional misuse category are also listed. In Table 6 it is clear that the code vulnerabilities have the most risk surfaces on blockchain. Under the code vulnerabilities, we divide codes into core software code which blockchain 1.0 and 2.0 are built upon, and the smart contract which only exists in blockchain 2.0. Under the core software code, we highlight the wallet security since quite a number of attacks hack the wallets.

5.2. Real attacks and bugs on blockchain systems

In this paper, we survey some real attacks and bugs on blockchain systems to raise awareness of the need for security on blockchain systems. Users use exchange platforms to make transactions on blockchain, and on blockchain a private key is kept in a digital wallet. Hence, exchange platforms and wallets are parts of blockchain systems.

5.2.1. Core software bug

Occurred in August 2010, the CVE-2010-5139 vulnerability was the most famous software bug in the Bitcoin network due to an integer overflow vulnerability in its protocol. Due to this bug, an invalid transaction of 0.5 BTC replaced with 184 trillion BTC was added in a normal block, and it took more than 8 h to resolve this problem [55]. In addition, when the version of Bitcoin upgraded from v0.7 to v0.8, there was a bug that a block processed in v0.8 was not processed in v0.7 because the database used BerkeleyDB in v0.8 and used LevelDB in v0.7. This bug caused the 6-h different blockchains to exist on the nodes with v0.8 and nodes with v0.7 [55].

5.2.2. Attacks related to cryptocurrency exchange platforms

In 2011, attackers took away several thousand BTC from Mt. Gox, a Tokyo-based Bitcoin exchange, due to deficiencies in network protocols, and in March 2014, another 650,000 BTC in its online coffers were stolen by hackers, which caused Mt. Gox to file for bankruptcy, due to a bug in the Bitcoin software to allow users to modify transaction IDs [56]. In December 2013, anonymous marketplace Sheep Marketplace had to shut down after it announced that one site vendor exploited a vulnerability and stole 5400 BTC [57]. In August 2016, hackers stole 119,756 BTC from the third-largest Bitcoin exchange, Bitfinex [58]. In July 2020, hackers hacked Cashaa, a UK-based cryptocurrency exchange, and stole 336+ BTC. In August 2020, hackers attacked a European cryptocurrency trading platform—2gether's servers and stole away 1.39 million USD [59].

5.2.3. Attacks with wallets

The user's wallet in the blockchain system stores his/her credentials and tracks digital assets associated with his/her address, the user credentials and any other information associated with his/her account. There have been some attacks in the past 10 years.

- Reported on September 5, 2012, Bitfloor, the fourth largest exchange dealing in US dollars, announced that hackers had hacked Bitfloor's server to access an unencrypted backup of the wallet keys and transfer away 24,000 BTC [60].

Table 6
Comprehensive blockchain security risk categories.

C1: Network Attacks	C2: Endpoint Security	C3: Intentional Misuse	C4: Code Vulnerabilities	C5: Data Protection	C6: Human Negligence
DoS, BGP (Border Gateway Protocol) hijacks, Routing attacks, Eclipse attack, Stealthier attacks, DNS attacks, Remote side-channel attacks	51% vulnerability, Sybil attacks, Personal key security, Mining malware, Cryptojacking Attacks	Injection, Insecure Deserialization, 51% vulnerability, Criminal activity, Double- spending, Selfish mining, Liveness attack, Balance attack, Timejacking attack, Finney attack, Race attack, SelfHolding attack	**Core software code (blockchain 1.0, 2.0)**: Injection, Using Components with Known Vulnerabilities, Security Misconfiguration, Broken Authentication, Broken Access Control, Insecure Deserialization, XSS,Transaction privacy leakage, Double-spending, Private key security **Wallet security** **Smart contract (blockchain 2.0)**: Vulnerabilities in smart contract, Criminal smart contracts, Under-priced operations, Under-optimized smart contract	Sensitive Data Exposure, Privacy leakage	Insufficient Logging & Monitoring, Security Misconfiguration

- On April 3, 2013, hackers hacked Instawallet, stole 35,000 BTC, and caused Instawallet to suspend operation indefinitely [61].
- On August 11, 2013, the Bitcoin Foundation announced that hackers exploited a generation bug of an old pseudo random number to enable them to solve the private key and steal balances from users' wallets [62].
- On both October 23 and October 26, 2013, an Australian Bitcoin bank was hacked, and all 4100 BTC held by the wallet service stored in the US server were stolen by hackers [63].
- Due to the multi-signature vulnerability in the Parity Wallet, a hacker stole 30 M from at least three Ethereum accounts by compromising their addresses on July 19, 2017 [64]. Unfortunately, the deployed new version of the Parity Wallet library contract had an undiscovered bug of not proper initialization at that time and caused another accident to be triggered on November 6, 2017, and the funds in affected multi-sig wallets were frozen [65].

5.2.4. Attacks and bugs with smart contract

One real instance of attacks on smart contracts is that when a specific smart contract DAO (Decentralized Autonomous Organization) built on Ethereum for the crowd-based venture capital fund, a hacker exploited its code weakness and stole more than 50 million USD worth of cryptocurrency reported on June 17, 2016 [66]. A hacker made use of sloppy smart contract coding to drain the funds in the smart contract [67]. On June 19, 2016, Vitalik Buterin listed categories of bugs with Ethereum contracts, including variable/function naming mix-ups, public data that should not have been public, reentrancy (A calling B calling A), sends failing due to 2300 gas limit, arrays/loops and gas limits, and subtle game-theoretic weaknesses [68].

In January 2017, there was Ether.Camp's Hacker Gold HKG in which a bug was discovered with the contract code that read "=+" instead of "+=" [69]. In October 2017, there was a 500,000 USD hack challenge from SmartBillions, and two hackers hacked and took away 400 ETH (120,000 USD) before the hackathon was stopped by SmartBillions [70]. In January 2018, a hacker discovered a bug of integer overflow with smart contracts using in Proof of Weak Hands (PoWH) coin and stole 888 ETH [71]. In October 2018, an attacker launched a reentrancy attack targeted at smart contracts of Spankchain and drained 165.38 ETH [72].

5.2.5. Network attacks

In August 2014, a research team in Dell SecureWorks Counter Threat Unit discovered that a BGP hijacker redirected the connections of cryptocurrency miners to a hijacker-controlled mining pool and obtained the miners' profit of an estimated 83,000 USD within four plus months [73]. In September 2016, a DDoS (Distributed DoS) attack was discovered to attack the Ethereum network such that an EXTCODESIZE opcode was called about 50,000 times per block by the attack transactions and hence greatly slowed down the network [74].

5.2.6. Endpoint attacks

Malware is one of the endpoint attacks. According to the report, malware infected more than one million computers, which were used by attackers to mine the 26+ million cryptocurrencies' tokens [75]. Cryptojacking is another endpoint attack, which cryptocurrency is mined in the web browser of the user while visiting a web. The attackers hacked and injected crypto mining scripts into Pirate Bay [76], CBS's Showtime [77] in 2017 and the Indian government web pages [78] in 2018 and gained the visitors' mining award by using the visitors' computers for mining. Attackers also injected cryptojacking code into third-party software (e.g., Google Tag Manager [79] and WordPress [80] in 2017, and Drupal [81] in 2018), and advertisements (e.g., YouTube ads [82] in 2018). Cryptojacking was also through 200,000 MikroTik routers infected by malware [83] in 2018, and corrupted Starbucks café's WiFi [84] at Buenos Aires in 2017 to let the infected computers mine the cryptocurrencies.

5.2.7. Attacks with IOTA

In January 2019, a hacker launched a phishing attack to collect the users' privacy keys for six months and then stole the users' IOTA worth 3.94 million USD [85]. At the same time, there was a DDoS attack on the IOTA network such that the IOTA developers were too busy to discover the hacker's theft activity [85]. In February 2020, to stop an attacker from stealing funds, the IOTA Foundation had to turn off the coordinator node for more than 12 days, which was responsible for confirming all transactions. The hackers broke IOTA own designed hash-function and could forge transactions [86].

Expanding from Hydra [87] and KEVM [88], we summarize attacks, attack years, categories based on Table 6, exploit values and root causes in Table 7. The total amount of exploit values at current BTC and ETH prices is more than 40 billion USD. Thus, the hackers have been and will continue being incentive to hack the blockchain systems to gain huge benefits.

Table 7
Attacks, years, categories, exploit values and root causes.

Attack	Year Category	Exploit Value	Root Cause
Mt. Gox	2011 C1	Several thousand BTC	Deficiencies in network protocols
Bitfloor	2012 C2	24,000 BTC (250,000 USD)	Bitfloor's server was hacked to leak an unencrypted backup of the wallet keys
Instawallet	2013 C4	35,000 BTC	Instawallet was hacked
Bitcoin Foundation	2013 C6	–	A generation bug with old pseudo random number
Sheep Marketplace	2013 C4	5400 BTC	One site vendor exploited a vulnerability
Mt. Gox	2014 C4	650,000 BTC (450 million USD)	A bug in software that allows users to modify transaction IDs
Dell SecureWorks	2014 C1	83,000 USD	BGP hijack
DAO	2016 C4	50 million USD	Code weakness: subtle game-theoretic weaknesses
Bitfinex	2016 C2 & C4	119,756 BTC (65 million USD)	Hackers stole BTC.
Ethereum network	2016 C1 & C4	–	DDoS attack: calling EXTCODESIZE opcode roughly 50,000 times per block
Gold HKG	2017 C4	–	A bug with contract code that reads "=+" instead of "+="
Parity Wallet	2017 C4	30 million USD	Addresses were comprised (Delegate call + exposed self-destruct)
SmartBillions	2017 C4	400 ETH (120,000 USD)	Broke into smart contract Broken caching mechanism
Parity Wallet	2017 C4	300 million USD	An undiscovered bug of not proper initialization (Delegate call + unspecified modifier)
Cryptojacking	2017–2018 C2 & C4	–	Hacked and inserted crypto mining script or cryptojacking code
PoWH	2018 C4	888 ETH	Integer overflow
Spankchain	2018 C4	165.38 ETH	Reentrancy attack
IOTA	2019 C2	3.94 million USD	A phishing attack to collect the users' privacy keys
IOTA	2020 C4	–	Custom-made hash-function was broken
Cashaa	2020 C2	More than 336 BTC	Suspect a piece of malware was installed onto the system
2gether	2020 C2	1.3 million USD	2gether's servers were hacked

Note: C1: network attacks, C2: endpoint security, C3: intentional misuse, C4: code vulnerabilities, C5: data protection, C6: human negligence; DAO: decentralized autonomous organization; PoWH: Proof of Weak Hands.

6. Security measures for blockchain

6.1. Security analysis

Smart Contract Bytecode Vulnerability Analysis. In 2016, Oyente was developed to find potential security bugs in smart contracts [89]. In 2018, Securify was presented as a security analyzer to automatically prove Ethereum smart contracts as unsafe/safe [90]. In 2018, ZEUS used symbolic model checking and abstract interpretation to validate the fairness and confirm the smart contracts' correctness, and about 94.6% of contracts were evaluated as vulnerable [91]. The well-known smart contract bytecode vulnerability analysis tools are listed in Table 8. Besides Oyente, Securify, and ZEUS, interested readers may find more detailed information about the analysis tools via their references.

In Table 8, all tools have detected certain vulnerabilities with a number of smart contracts, although some tools detect more vulnerabilities and/or detect more vulnerable contracts. In other words, the developers should pay great attention to designing smart contracts against known or unknown attacks since not all contracts are secure enough. The other features of individual tools are also listed in the table so as to facilitate the users to know more about the analysis tools for smart contracts.

Transactions and Transaction Logs analysis. In 2020, TxSpector [99] was the first generic framework to perform bytecode-level, logic-driven analysis on Ethereum transactions for attack detection, such as Reentrancy, UncheckedCall, Suicidal Vulnerability, Timestamp Dependence, Misuse-of-Origin, Failed Send, Mishandled Exception, Unsecured Balance, and DoS. Based on the transaction logs, an Ever-evolving Game was presented also in 2020 to analyze attacks in the real world and defenses adopted in the wild [100].

Honeypot Smart Contracts. Instead of exploiting the vulnerabilities of smart contracts, hackers developed honeypot smart contracts with hidden traps, and $H_{ONEY}B_{ADGER}$ was developed in 2019 to analyze more than 2 million smart contracts and identify 690 honeypot smart contracts [101].

Consensus Algorithm Analysis. In 2016, a group of researchers from ETH Zurich and NEC Laboratories presented a framework to quantitatively analyze the PoW's security and performance [102]. In 2019, Zhang and Preneel evaluated and showed that PoW could not achieve the ideal chain quality and could not be resistant to attacks of selfish mining, double-spending, and feather-forking [103].

6.2. Detecting malicious codes & bugs

In 2018, Jiang et al. proposed ContractFuzzer to fuzz smart contracts to detect vulnerability [104], Liu et al. presented ReGuard of a fuzzing-based analyzer in their demo paper to automatically detect the reentrancy bugs of the most common bug type in the smart contracts [105], and Hydra was developed by Breidenbach et al. to use bug bounties to enable rewarding of critical bugs and runtime detection [87]. In 2019, EVMFuzzer was proposed to use a differential fuzzing technique by continuously generating seed contracts as input to the target EVM and based on the execution results to detect vulnerabilities in the EVM [106]. In 2020, a lightweight test-generation approach—HARVEY was presented to effectively detect security vulnerabilities and bugs for smart contracts [107].

6.3. Core software codes security

In 2017, SmartPool as a decentralized mining pool was designed to prevent the phenomenon that close to 80% of Ethereum's and 95% of Bitcoin's mining power resided with less than six and ten mining pools, respectively [108]. In 2019, Drijvers et al. pointed out subtle flaws with the two-round multi-signature scheme and then proposed mBCJ as a provably secure yet highly efficient alternative [109]. In 2020, Drijvers et al. presented Pixel, a pairing-based forward-secure multi-signature scheme, against posterior corruption attacks [110], and Sun et al. presented Counter-RAPTOR to mitigate and detect active routing attacks [111].

6.4. Secure smart contract

In 2016, Luu et al. presented methods to enhance Ethereum operational semantics to reduce the smart contracts' vulnerabilities [92]. In

Table 8
Smart contract bytecode vulnerability analysis tools and feature comparison.

Smart Contract Analysis Tools	Analysis Domain	Vulnerabilities Detected	Number of Smart Contracts Analyzed	Number of Vulnerable Smart Contracts	Remark
Oyente [92]	Symbolic execution	Timestamp dependence, transaction-ordering dependence, mishandled exceptions, reentrancy	19, 366	8, 833	The first symbolic execution-based tool
Mythril [93]	Symbolic execution	Integer underflows, owner-overwrite-to-ether-withdrawal, and others	Unknown	Have, but no given number	
teEther [94]	Symbolic execution	Erroneous visibility, erroneous constructor, semantic confusion, logic flaws, inter-contract exploits	38,757	815	
Manticore [95]	Symbolic execution	Unprotected function, integer-overflow, undefined behaviour, misconfiguration, numeric, timing, business logic.	100	Have, but no given number	
ZEUS [91]	Abstract interpretation	Reentrancy, failed send, unchecked send, integer overflow/ underflow, transaction state dependency, incorrect logic, absence of logic, block state dependency, logically correct but unfair, transaction order dependency	22, 493	21, 281 (94.6%)	Check smart contracts written in Solidity against a user-defined policy
M_{AIAN} [96]	Symbolic execution	Entire contract execution traces, i.e., leaky contracts, prodigal contracts, suicidal contracts, greedy contracts	970,898	34,200	Detecting across a long invocation sequence a smart contract
Securify [90]	Data-flow analysis	Ether liquidity, unrestricted write, no writes after calls, restricted transfer, mishandled exception, transaction ordering dependency, unexpected arguments	Ethereum Virtual Machine (EVM): 24, 594; Solidity dataset: 100	6.50%	Explore all contract behaviours
Vandal [71]	Abstract interpretation	Reentrancy, unsecured balance, use of ORIGIN, destroyable contract, unchecked send	141,000	Have, but no given number	Convert bytecode to semantic logic relation
MadMax [97]	Abstract interpretation	Unbounded mass operations, integer overflows, non-isolated external calls in wallet griefing, incentive attacks	6.33 million	5.42%	A tool to find gas-based vulnerabilities
Osiris [98]	Symbolic execution	Integer bugs: truncation bugs, arithmetic bugs, and signing-related bugs	1.2 million	42,108	
E_{TH}BMC [94]	Symbolic execution	Extract ether, redirect control flow, self-destruct contract, parity vulnerability, more exploits	roughly 2.2 million	5905	More precisely reasoning of EVM internals

2016, Town Crier was developed to ensure only authenticated data be input into the smart contracts [112]. In 2018, FSolidM was presented as a tool to enable the developers to define secure smart contracts as FSMs (finite state machines) and enhance security and functionality [113], and Arbitrum was designed to verify off-chain what a VM would do so as to improve scalability and privacy [114]. In 2020, a research group from Korea University described VERISMART to ensure arithmetic safety to address security concerns of Ethereum smart contracts [115].

6.5. Smart contract verification

In 2018, Amani et al. created a program logic at the bytecode level to extend an existing EVM formalisation so as to formally verify EVM smart contracts [116], and a formal modeling approach was proposed by Abdellatif & Brousmiche to verify the blockchain and users' behavior of the smart contract [117]. In 2020, Sun & Yu established a framework to verify the security vulnerabilities of smart contracts, e.g., the Binance Coin (BNB) contract [118], and Permenev et al. presented VerX to verify the functional properties of smart contracts on Ethereum automatically [119].

6.6. Privacy preserving

In 2016, Hawk was developed to protect the privacy of transactions without storing the clear text on the blockchain via a private smart contract [120]. In 2018, Obscuro was presented to provide a secure and efficient Bitcoin mixer so that payers and payees could not be linked together to achieve anonymous payments [121]. In 2019, Ouroboros Crypsinous was described to analyze the privacy-preserving PoS protocol [122], and BITE was developed to enable the privacy preserving requests from light clients [123]. In 2020, Zexe was demonstrated to achieve privacy-preserving analogues of some popular applications [124]. In 2020, remote side-channel attacks were presented on receiver privacy [125].

6.7. Monitor and regulations agaist hackers' wallets

Cryptocurrency exchange platforms may lock any funds that were from the hacked wallet. New regulations on anti-money laundering (AML) are enforced to make it difficult for hackers to move the funds [126].

6.8. Hard fork

To respond to the hacking of the DAO, Ethereum was divided into Ethereum Classic and the new Ethereum. As a hard fork from the original software, the new Ethereum can protect against further malware attacks. The Ethereum Classic has tokens called ETC while the new Ethereum has tokens called ETH. Both the new Ethereum and Ethereum Classic have a common ancestry prior to Block 1,920,000.

7. Challenges and research trends

There are some existing surveys that have presented the future trends or scopes for blockchain technology. Blockchain testing, big data analytics, blockchain applications, smart contracts, stopping the tendency to centralization, and artificial intelligence are listed by the same research group in Refs. [3,27]. A hybrid consensus mechanism, more efficient consensus, code obfuscation, execution trusted computing against privacy leakage risks, application hardening, and an efficient data cleanup & detection mechanism are presented in Ref. [29]. A standard testing mechanism, big data analytics, smart contract development and evaluation are proposed in Ref. [45]. Resolving bugs in blockchain technology, more use cases and applications, and promoting the awareness of blockchain technology are described in Ref. [44]. Besides those valid trends and scopes, this paper would highlight the below challenges and research trends.

7.1. Scalability

Scalability on Transactions. In Table 1, the maximum TPS (transactions per second) is from 27 of Bitcoin to EOS of 3996. PoW is capable of processing anywhere between 10 TPS and 27 TPS worldwide. Ethereum 2.0 will upgrade and switch to the more efficient protocol PoS to make Ethereum more scalable, and will support 1000s of TPS [127]. A few delegates in EOS that use the DPoS consensus algorithm have the right to vote and validate blocks, and hence EOS is more centralized and is easier for some delegates to combine together to launch 51% attacks. The communication cost in PBFT quickly grows if the number of nodes increases, and hence it is suitable for private setups without a large number of nodes, but with many transactions. Currently, Hyperledger Fabric based on PBFT achieves about 3500 TPS. Hyperledger Sawtooth based on PoET achieves 2300 TPS.

In 2019, Perun was proposed as an off-chain payment channel system instead of on-chain transactions to increase TPS [128], and a sidechain construction was provided for PoS sidechain systems to enable scalability [129]. In 2020, Yu et al. proposed O_{HIE} as a permissionless protocol to improve the transaction throughput to 4–10 Mbps [130]. Currently, Ethereum and Bitcoin process only about 5 KB or 10 TPS on average. So O_{HIE} can achieve 8000–20,000 TPS. On the other hand, Visa's payment network can process over 65,000 TPS, as stated in August 2017 [131]. Thus, the scalability of blockchain in terms of TPS in a real distribution environment is still an outstanding challenge.

Scalability on Chain Data Sharing. The block sizes for Bitcoin, Bitcoin Cash, and Ethereum are 1 MB, between 8 MB and 32 MB, and under 60 KB, respectively. IBM blockchain supply chain solutions [132] and VeChain [133] record the shared data on the blockchain which limit the scalability of their solutions. A large number of stakeholders may be involved, and the data that need to be shared among the stakeholders could be massive and not limited to logistic data. As more stakeholders join and the shared data grows, the on-chain data sharing system will be in danger of scalability issues.

To increase the scalability and also take advantage of blockchain technology, the data can be shared on an off-chain dedicated channel and the link or even proof of the data sharing can be recorded in the blockchain for tracing and auditing. Off-chain data sharing solutions require inter-company channels, which increase the company's burden for building and maintaining these channels. In addition, these solutions cannot guarantee the integrity of the data shared by a company. For example, Company A can tamper with the original data to make the data meet Company B's specific requirements and then share the data with Company B. To decrease the burden of the involved companies, the data can be positioned and shared on a cloud platform. We have proposed one technology in this area of a blockchain-based access control and data sharing framework for supply chains, which can be referred to in our patent filed document [134].

7.2. Securer software codes

From Table 8, we can know that almost every year attacks on software code and smart contracts have happened. Security is a non-negotiable aspect of any asset-related software. Smart contract security is a high requirement because smart contract deals with valuable information, e.g., cryptocurrencies, tokens, and other digital assets. The transactions built with smart contracts are irreversible, and the software codes of smart contracts are very difficult to be modified or patched if a bug is discovered [135]. A few constraints on smart contracts are in place to secure the blockchain environment from attackers. Additional to the accounts and transactions being immutable and secured through the cryptographically hashed chains, for example, EOS faces the challenge of securing the smart contract execution to withstand malicious attacks [136]. In 2020, there is a research work on Flash Boys 2.0 continuing to show the risks of smart contracts that the arbitrage bots and miners extractable value of transaction-ordering dependencies in smart

contracts pose a realistic threat to Ethereum [137]. According to Ref. [135], it is very hard to assure the security of smart contract code, and hence guaranteeing the security of smart contracts is one of the outstanding challenges for blockchain.

7.3. Audit, zero trust & anomaly detection

Smart Contract Audit. Before the deployment of the smart contract, one further step is to audit the smart contract. In 2018, Erays was presented to reverse engineering the smart contract into high level pseudocode and then manually analyze several contract properties [138]. One of the research trends could be to further develop an audit tool to automatically audit more or all properties of smart contracts.

Zero Trust Network Access for Endpoint Security. Table 7 also clearly shows the critical importance of endpoint security, including the server security which needs to safely guard the users' credentials, ensure the wallet security, harden the server protection, and prevent phishing attacks, insider attacks, and other unknown attacks. Thus, zero trust network access that continues authenticating the endpoints is one of the research trends.

Monitor and Anomaly Detection. Network monitoring and attack/anomaly detection is a continuous effort for blockchain security. Machine learning, deep learning, and federate learning on analyzing transactions, logs, behaviors, and data besides the existing parsing approach [139] would be one of the research trends to secure the blockchain systems. ETH-EDS in 2020 used random forest classification to detect eclipse attacks [140], which is one example to use machine learning technology for attack detection.

7.4. Privacy preserving

With more and more data stored on the blockchain, a concern from the organization and individuals is privacy leakage. Some techniques of code obfuscation, homomorphy encryption, trusted executing platform (e.g., Intel SGX), and smart contract for privacy preserving would be promising directions.

7.5. Quantum computing impact on blockchain

ECDSA. In ECDSA, used for signing transactions in blockchain, a public key is calculated from its private key, with a one-way function that is easy to compute the public key in one direction of Elliptic curve multiplication, but it is impossible to reverse engineering to do the division to get the private key because of the hardness of solving mathematical discrete logarithm problems, which assumes that an astronomical amount of time is required to solve and is hence not practical. Therefore, the users in blockchain can sign the digital signature with their private key to show their ownership.

IBM, Intel, Google, Rigetti, D-Wave, IonQ, Microsoft, and major nation-states are actively involved in research and developing quantum computing. In 1994, a quantum algorithm published by Shor can break the security assumptions of the most common algorithms of public key cryptography [141] and an improved Shor's algorithm is the potential to break ECDSA [142].

Ethereum developers are testing the new quantum-resistance signature algorithms, such as XMSS, hash ladder signatures, and SPHINCS, and the Ethereum 2.0 Serenity update will replace the ECDSA scheme. Post-quantum algorithms will be still hard problems for quantum computers. The National Institute of Standards and Technology (NIST) is processing and standardizing public key cryptographic algorithms with quantum-resistance. In July 2020, NIST selected 15 algorithms from 26 post-quantum cryptography algorithms in the second-round list, and now those 15 algorithms are in the third round of public review [143].

Address. Hash function's preimage resistance makes sure that given the P2PKH address, it is mathematically impossible to reverse-engineer its public key. If its public key is unknown, the quantum computer cannot derive its private key. However, once any amount of funds is transferred from a particular P2PKH address, its public key will be disclosed to verify its transaction digital signature, and hence its private key is no longer secure under quantum computing. The worst situation is that a recipient's public key is directly used as the Bitcoin address called ‘pay to public key’ (P2PH). An analysis showed that about 25% of all Bitcoins (over 4 million BTC) have the potential to suffer a quantum attack [144].

The blockchain community shall also address the quantum computing impact on blockchain. Only post-quantum cryptography is resistant to quantum attacks. One of the research trends is to investigate how to apply post-quantum cryptography to building the robust and quantum-resistant blockchain. It will then have to hard fork the blockchain, e.g., blockchain 3.0, which implements the new post-quantum cryptography protocol and is different from the current blockchain.

7.6. IOTA security

As Bitcoin and Ethereum-based cryptocurrencies encounter the problems of scalability and transaction fees, IOTA may be a good alternative due to its very different nature structure of vertices and edges in using of DAGs instead blocks + chain. With Tangle technology, IOTA claims to be very scalable without a limit and charges zero transaction fees. However, Tangle technology faces some concerns of not being able to store the transactions' orders properly [145] and vulnerabilities with their own designed IOTA hash function called Curl. IOTA needs to overcome these challenges. When the technology is mature, it will be expected to widespread adoption in the industry of IoT, a rapidly growing and huge potential area.

7.7. Regulation and standard issue

First, it is expected that cryptocurrencies are getting popular, which will create convenience and save the cost for the fund transactions. On the other hand, it also weakens the countries' financial policies and control. Second, more international blockchain applications are emerging. For example, blockchain systems are used to verify the COVID-19 vaccine injection certificates. Hence, there is a need to have the regulations and agreements among different countries to mutually accept the injection certificates stored on the blockchain systems. Third, even within the same country, multiple parties shall agree to use the blockchain as a common infrastructure, which could be a big challenge, not to mention making a common or international standard. Thus, regulation and standards will be one of the challenges for the massive deployment of blockchain systems.

8. Related work

There are some survey papers about blockchain. In January 2017, Sankar et al. described three broad types of blockchains, and analyzed and compared qualitatively three consensus algorithms, namely Stellar consensus protocol, Corda, and Hyperledger Fabric [26]. In June 2017, Zheng et al. surveyed the blockchain architecture, including types of blockchain, compared consensus algorithms qualitatively, and presented the vulnerabilities of privacy leakage and selfish mining and migration solutions [27]. In August 2017, Ji.H. Park and Jo.H. Park surveyed about blockchain structure and Bitcoin, presented the security challenges including the majority of attacks (51% attacks), security of the transaction, security of software and security of wallet, and adapt blockchain security to cloud computing [55]. Another work available online in August 2017 conducted a survey on blockchain security about the security risks, real attacks, and academic security enhancements till 2017. In September 2017, Lin and Liao presented security issues of 51% attacks and some challenges, including fork problem, data synchronization and confirmation time, regulations, and integration cost problems [2].

In May 2018, a work from Kennesaw State University presented the use of blockchain and cryptography to ensure data confidentiality,

Table 9
Summaries of various survey works.

Work	Blockchain Category	Consensus Protocols		Applications	Scalability	Blockchain Security	Quantum Computing	Future Direction
		Qualitative Comparison	Quantitative Comparison					
[26]	Yes	Yes						
[27]	Yes	Yes			Yes	Partial		Yes
[55]						Partial		
[2,28]	Yes	Yes		Yes	Yes	Partial		
[3]	Yes	Yes	Partial	Yes	Yes	Partial		Yes
[150]				Yes				
[29]		Yes				Partial		Yes
[30]		Yes						
[31]	Yes	Yes	Partial		Yes	Partial		
[32]				Yes				
[45]	Yes	Yes	Partial	Yes	Yes			Yes
[44]				Yes		Partial		Yes
[146]	Yes	Yes		Yes		Partial		
[147]		Yes	Partial			Partial		Yes
[148]				Yes	Yes	Surveyed on vulnerabilities	Yes	Yes
[149]		Yes		Yes	Yes	Examined security in process, data and infrastructure levels	Yes	Yes
This paper	Yes	Yes	As many as possible	Yes	Yes	Comprehensive blockchain security risk categories, real attacks, bugs & root causes, recent security measures	Yes	Yes

authenticity, integrity, and privacy preserving for various blockchain applications, instead of security on blockchain itself [28]. In October 2018, Zheng et al. conducted a survey on blockchain technology which included consensus algorithms, applications, challenges on scalability, privacy leakage, selfish mining, and future directions on blockchain testing, big data analytics, stopping the tendency to centralization, smart security analysis, and artificial intelligence [3]. In November 2018, challenges and security with blockchain were surveyed by Tunisia researchers [146]. In December 2018, Chen et al. surveyed only blockchain applications in different domains [43].

In August 2019, Monrat et al. conducted a survey on blockchain architecture, including transaction process, block structure and characteristics of blockchain, category of blockchain, consensus procedures, blockchain applications, trade-offs, and the future scope of blockchain technology [45]. In November 2019, Dave et al. surveyed the implementations of blockchain technology in the agricultural sector, education sector, supply chain management, healthcare industry, etc. [44]. In March 2020, Aguiar et al. surveyed and used blockchain technology to boost healthcare security and reliability and enhance patient privacy [30]. One survey work received in December 2019 and published in April 2020 presented the blockchain technology, applications, and issues including scalability, nothing-at-stake, etc. [31]. In 2020, Saad et al. presented a systematical overview of the blockchain attack surface [147]. In January 2021, Berdik et al. presented their survey paper on blockchain to ensure information integrity and security [32].

There are some survey papers on blockchain security. In 2019, Dasgupta et al. surveyed the potential vulnerabilities of blockchain and showed blockchain development trends [148]. In 2020, Leng et al. examined blockchain security from the process level, the data level, and the infrastructure level to identify the research gap and suggest future directions of research in blockchain security [149].

Table 9 summarizes the related survey work and our work in this paper. It is also clear to show our contributions in this paper. First, we provide as many quantitative comparisons on consensus algorithms as possible, while others only provide partial comparisons. Second, the security of blockchain itself is a focus in this paper, which the majority of previous surveys only partially presented or did not present, and some survey papers on blockchain security surveyed the potential vulnerabilities and examined security in the process, data, and infrastructure levels, respectively. In our paper, we assess the blockchain security from risk analysis to derive comprehensive blockchain security risk categories, analyze the real attacks and bugs against blockchain and root causes, and present the recently developed security measures on blockchain. Last but not least, Table 9 shows that other survey papers cover 2 to 7 areas, respectively, while our work consists more comprehensive survey on 8 areas of blockchain.

9. Conclusion

This paper has first conducted a deeper survey on blockchain technology in terms of overview, consensus algorithms, smart contracts, and cryptography for blockchain. It presented the history of blockchain, and compared the five most common consensus algorithms and one most different consensus algorithm in as much detail and quantitative as possible. Public key cryptography, Zero-Knowledge Proof, and hash functions used in blockchain have been described in detail for integrity, authentication, nonrepudiation, and payment address required in blockchain systems. This paper has then listed the comprehensive applications of blockchain. It has further presented the rich information and comparisons of eight cryptocurrencies as the first blockchain application, supply chain as a widely used case, and Smart Dubai Office as the first whole government service application. Further, the security of blockchain itself is a focus in this paper. It has described the comprehensive security risks categories based on the Top 10 Web Application Security Risks, low level risks, and high level risks. It has surveyed many real attacks and bugs on blockchain systems and listed out their root causes. The paper has then presented the security measures in the areas of security analysis, detecting malicious codes & bugs, software codes security, privacy preserving, and so on. Specially, it has presented and compared eleven smart contract bytecode vulnerability analysis tools. Finally, the challenges and research trends have been presented to build more scalable and securer blockchain systems for massive deployments.

We hope that our effort will help someone understand blockchain technology and blockchain security issues. The users who use blockchain to do the transactions will pay more attention to the security of blockchain itself. We also expect that the researchers will benefit from our study for their further research in developing blockchain technology and addressing blockchain security issues.

Conflict of 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] M.S. Ali, M. Vecchio, M. Pincheira, et al., Applications of blockchains in the internet of things: a comprehensive survey, IEEE Communications Surveys and Tutorials 21 (2) (2019) 1676–1717.
[2] I.-C. Lin, T.-C. Liao, A survey of blockchain security issues and challenges, Int. J. Netw. Secur. 19 (5) (2017) 653–659.
[3] Z. Zheng, S. Xie, H.-N. Dai, X. Chen, H. Wang, Blockchain challenges and opportunities: a survey, Int. J. Web Grid Serv. 14 (4) (2018) 352–375.
[4] D. Chaum, Computer Systems Established, Maintained, and Trusted by Mutually Suspicious Groups, Ph.D Thesis, University of California, Berkeley, CA, USA, 1982.
[5] S. Haber, W.S. Stornetta, How to time-stamp a digital document, J. Cryptol. 3 (2) (1991) 99–111.
[6] D. Bayer, S. Haber, W.S. Stornetta, Improving the efficiency and reliability of digital time-stamping, in: R. Capocelli, A. De Santis, U. Vaccaro (Eds.), Sequences II, Springer, New York, NY, USA, 1993.
[7] R. Sharma, Bit gold, Investopedia, 2021. Available online: https://www.investopedia.com/terms/b/bit-gold.asp. (Accessed 24 October 2021).
[8] S. Nakamoto, Bitcoin: a peer-to-peer electronic cash system. https://bitcoin.org/bitcoin.pdf, October 2008.
[9] R. Sheldon, A timeline and history of blockchain technology. https://whatis.techtarget.com/feature/A-timeline-and-history-of-blockchain-technology, 2021.
[10] V. Buterin, Ethereum whitepaper. https://ethereum.org/en/whitepaper/, 2013.
[11] A. Groetsema, A. Groetsema, N. Sahdev, N. Salami, R. Schwentker, F. Cioanca, Blockchain for Business: an Introduction to Hyperledger Technologies, The Linux Foundation, 2019.
[12] P. Vasin, BlackCoin's Proof-of-Stake Protocol v2. https://blackcoin.org/blackcoin-pos-protocol-v2-whitepaper.pdf. Accessed March 21, 2021.
[13] Crushcrypto, WHAT IS DELEGATED proof-OF-STAKE? Crushcrypto (2018). Available online: https://crushcrypto.com/what-is-delegated-proof-of-stake/. (Accessed 21 March 2021).
[14] Intel Corporation, PoET 1.0 Specification, 2016. Available online: https://sawtooth.hyperledger.org/docs/core/releases/1.0/architecture/poet.html. (Accessed 21 March 2021).
[15] M. Castro, B. Liskov, Practical Byzantine Fault tolerance, in: Proceedings of the Third Symposium on Operating Systems Design and Implementation; 22–25 Feb 1999; New Orleans, LA, USA, USENIX Association, Berkeley, CA, USA, 1999, pp. 173–186.
[16] S. Popov, The Tangle. https://whitepaper.io/document/3/iota-whitepaper, 2018. Accessed March 21, 2021.
[17] Academy Binance, What is a Directed Acyclic Graph (DAG) in Cryptocurrency? Academy Binance, 2020. Available online: https://academy.binance.com/en/articles/what-is-a-directed-acyclic-graph-dag-in-cryptocurrency. (Accessed 29 April 2021).
[18] OpenEthereum, Proof of Authority Chain. https://openethereum.github.io/Proof-of-Authority-Chains. Accessed March 21, 2021.
[19] J. Kwon, Tendermint: Consensus without Mining. https://tendermint.com/static/docs/tendermint.pdf, 2014. Accessed March 21, 2021.
[20] B. Chase, E. MacBrough, Analysis of the XRP ledger consensus protocol, arXiv, 2018, preprint.
[21] L. Luu, V. Narayanan, K. Baweja, et al., SCP: a computationally-scalable byzantine consensus protocol for blockchains, IACR Cryptology ePrint Archive, 2015, p. 1168.
[22] M. Ghosh, M. Richardson, B. Ford, R. Jansen, A TorPath to TorCoin: Proof-of-Bandwidth Altcoins for Compensating Relays, 2021. https://dedis.cs.yale.edu/dissent/papers/hotpets14-torpath.pdf. Accessed March 21, 2021.
[23] NEM, NEM Technical Reference. https://nemplatform.com/wp-content/uploads/2020/05/NEM_techRef.pdf, 2018. Accessed March 21, 2021.
[24] K. Karantias, A. Kiayias, D. Zindros, Proof-of-Burn, in: J. Bonneau, N. Heninger (Eds.), Financial Cryptography and Data Security. FC 2020. Lecture Notes in Computer Science, vol. 12059, Springer, Cham, 2020, pp. 523–540.
[25] A. Hayes, Proof of Capacity (cryptocurrency), Invest, 2020. Available online: https://www.investopedia.com/terms/p/proof-capacity-cryptocurrency.asp. (Accessed 21 March 2021).
[26] L.S. Sankar, S. M, M. Sethumadhavan, Survey of consensus protocols on blockchain applications, in: 2017 International Conference on Advanced Computing and Communication Systems (ICACCS -2017); 6–7 Jan 2017; Coimbatore, India, IEEE, Piscataway, NJ, USA, 2017, pp. 1–5.
[27] Z. Zheng, S. Xie, H. Dai, et al., An overview of blockchain technology: architecture, consensus, and future trends, in: IEEE 6th International Congress on Big Data; 25–30 Jun 2017; Honolulu, HI, USA, IEEE, Piscataway, NJ, USA: IEEE, 2017, pp. 557–564.
[28] A.P. Joshi, M. Han, Y. Wang, A survey on security and privacy issues of blockchain technology, Mathematical Foundations of Computing 1 (2) (May 2018) 121–147.
[29] X. Li, P. Jiang, T. Chen, X. Luo, Q. Wen, A survey on the security of blockchain systems, Future Generat. Comput. Syst. 107 (June 2020) 841–853.
[30] E.J. De Aguiar, B.S. Faiçal, B. Krishnamachari, J. Ueyama, A survey of blockchain-based strategies for healthcare, ACM Comput. Surv. 53 (2) (2021) 1–27.
[31] H.T.M. Gamage, H.D. Weerasinghe, N.G.J. Dias, A survey on blockchain technology concepts, applications, and issues, SN Computer Science 1 (114) (2020).
[32] D. Berdik, S. Otoum, N. Schmidt, D. Porter, Y. Jararweh, A survey on blockchain for information systems management and security, Inf. Process. Manag. 58 (1) (January 2021).
[33] S. King, S. Nadal, PPCoin: peer-to-peer crypto-currency with proof-of-stake. https://decred.org/research/king2012.pdf, August, 2012.
[34] D. Schmidt, Delegated proof of stake. https://www.benzinga.com/money/delegated-proof-of-stake/, July, 2020.
[35] J. Frankenfield, Proof of Elapsed Time (PoET) (Cryptocurrency), Invest, October 16, 2020. Available online: https://www.investopedia.com/terms/p/proof-elapsed-time-cryptocurrency.asp. (Accessed 21 March 2021).
[36] IOTA. https://www.iota.org/. Accessed March 23, 2021.
[37] T. Kozak, Consensus Protocols that Meet Different Business Demands, Part I, Intellectsoft, 2018. Available online: https://blockchain.intellectsoft.net/blog/consensus-protocols-that-meet-different-business-demands/. (Accessed 21 March 2021).
[38] CrushCrypto, WHAT IS proof OF WORK?. https://crushcrypto.com/what-is-proof-of-work/, 2021.
[39] CrushCrypto, What is practical byzantine fault tolerance (PBFT)? Crushcrypto, 2018. Available online: https://crushcrypto.com/what-is-practical-byzantine-fault-tolerance/. (Accessed 21 March 2021).
[40] S. Zhang, J.-H. Lee, Analysis of the main consensus protocols of blockchain, ICT Express 6 (2020) 93–97.
[41] R. Santos, K. Bennett, E. Lee, Blockchain: Understanding its Uses and Implications, The Linux Foundation, 2021. Available online: https://www.edx.org/course/blockchain-understanding-its-uses-and-implications. (Accessed 5 October 2021).
[42] A.M. Antonopoulos, Mastering Bitcoin, second ed., O'Reilly Media, Inc., Sebastopol, CA, USA, June 2017.
[43] W. Chen, Z. Xu, S. Shi, Y. Zhao, J. Zhao, A survey of blockchain applications in different domains, in: International Conference on Blockchain Technology and Applications (ICBTA); 10–12 Dec; Xi'an, China, ACM, New York, NY, USA, 2018, pp. 17–21.
[44] D. Dave, S. Parikh, R. Patel, et al., A survey on blockchain technology and its proposed solutions, Procedia Comput. Sci. 160 (2019) 740–745.
[45] A.A. Monrat, O. Schelen, K. Andersson, A survey of blockchain from the perspectives of applications, challenges, and opportunities, IEEE Access 7 (2019) 117134–117151.
[46] W. Meng, E.W. Tischhauser, Q. Wang, Y. Wang, J. Han, When intrusion detection meets blockchain technology: a review, IEEE Access 6 (2018) 10179–10188.
[47] Cryptoslate, Coin rankings. https://cryptoslate.com/coins/, February 28, 2021.
[48] L. Conway, The 10 most important cryptocurrencies other than bitcoin, Invest (Jun 1, 2021).
[49] S. Kovach, Tesla Buys $1.5 Billion in Bitcoin, Plans to Accept it as Payment, CNBC, February 8, 2021. Available online: https://www.cnbc.com/2021/02/08/tesla-buys-1point5-billion-in-bitcoin.html. (Accessed 23 March 2021).
[50] J. Leng, P. Jiang, K. Xu, Q. Liu, J.L. Zhao, Y. Bian, R. Shi, Makerchain: a blockchain with chemical signature for self-organizing process in social manufacturing, J. Clean. Prod. 234 (2019) 767–778.
[51] J. Leng, S. Ye, M. Zhou, J.L. Zhao, Q. Liu, W. Guo, W. Cao, L. Fu, Blockchain-secured smart manufacturing in industry 4.0: a survey, IEEE Transact. Syst. Man Cybernet.: Systems 51 (1) (2021) 237–252.
[52] J. Leng, G. Ruan, P. Jiang, K. Xu, Q. Liu, X. Zhou, Blockchain-empowered sustainable manufacturing and product lifecycle management in industry 4.0: a survey, Renew. Sustain. Energy Rev. 132 (2020), 110112.
[53] J. Leng, D. Yan, Q. Liu, K. Xu, J.L. Zhao, R. Shi, L. Wei, D. Zhang, X. Chen, ManuChain: combining permissioned blockchain with a holistic optimization model as Bi-level intelligence for smart manufacturing, IEEE Transact. Syst. Man Cybernet.: Systems 50 (1) (2020) 182–192.
[54] H. Poston, Mapping the OWASP top ten to blockchain, Procedia Comput. Sci. 177 (2020) 613–617.
[55] Ji.H. Park, Jo.H. Park, Blockchain security in cloud computing: use cases, challenges, and solutions, Symmetry 9 (8) (2017) 164.
[56] J. Frankenfield, Mt. Gox, Investopedia, 2021. Available online: https://www.investopedia.com/terms/m/mt-gox.asp. (Accessed 26 March 2021).
[57] E. Yu, Anonymous Website Disappears with $100M in Bitcoin, ZDNet, December 5, 2013. Available online: https://www.zdnet.com/article/anonymous-website-disappears-with-100m-in-bitcoin/. (Accessed 26 March 2021).
[58] J. Horwitz, I. Kar, One of the World's Largest Bitcoin Exchanges Lost $65 Million in a Hack, QUARTZ, August 3, 2016. Available online: https://qz.com/748995/one-of-the-worlds-largest-bitcoin-exchanges-lost-65-million-in-a-hack/. (Accessed 26 March 2021).
[59] F. Erazo, Hackers Steal Over $1.3M from European Crypto Trading Platform, Cointelegraph, Aug 03, 2020. Available online: https://cointelegraph.com/news/hackers-steal-over-13m-from-european-crypto-trading-platform. (Accessed 26 March 2021).
[60] V. Buterin, Bitfloor Hacked, $250,000 Missing, Bitcoin Magazine, September 5, 2012. Available online: https://bitcoinmagazine.com/business/bitfloor-hacked-250000-missing-1346821046. (Accessed 26 March 2021).
[61] C.K. Elwell, M.M. Murphy, M.V. Seitzinger, Bitcoin: Questions, Answers, and Analysis of Legal Issues, Congressional Research Service, July 15, 2014. Available online: https://www.everycrsreport.com/reports/R43339.html. (Accessed 26 March 2021).
[62] R. Chirgwin, Android Bug Batters Bitcoin Wallets, The Register, August 12, 2013. Available online: https://www.theregister.com/2013/08/12/android_bug_batters_bitcoin_wallets/. (Accessed 26 March 2021).
[63] B. Grubb, Australian Bitcoin Bank Hacked: $1m+ Stolen, Brisbane Times, November 8, 2013. Available online: https://www.brisbanetimes.com.au/technology/australian-bitcoin-bank-hacked-1m-stolen-20131108-hv2iv.html. (Accessed 26 March 2021).
[64] W. Zhao, $30 Million: Ether Reported Stolen Due to Parity Wallet Breach, Coindesk, Jul 20, 2017. Available online: https://www.coindesk.com/marke

ts/2017/07/19/30-million-ether-reported-stolen-due-to-parity-wallet-breach/. (Accessed 26 March 2021).
[65] Parity Technologies, Security Alert. https://www.parity.io/security-alert-2/, November 08, 2017. Accessed April 11, 2021 from.
[66] N. Popper, A Hacking of More than $50 Million Dashes Hopes in the World of Virtual Currency, N. Y. Times, June 17, 2016.
[67] A. Lewis, A Gentle Introduction to Ethereum, Bits on Blocks, October 2, 2016.
[68] V. Buterin, Thinking about Smart Contract Security, ethereum foundation blog, June 19, 2016. Available online: https://blog.ethereum.org/2016/06/19/thinking-smart-contract-security/. (Accessed 11 April 2021).
[69] Spartak_t, HackerGold (HKG) has a SERIOUS bug. https://bitcointalk.org/index.php?topic=1744115.0, January 08, 2017. Accessed April 11, 2021.
[70] J. Solana, \500K hack challenge backfires on blockchain lottery SmartBillions, Available online: https://calvinayre.com/2017/10/13/bitcoin/500k-hackchallenge-backfires-blockchain-lottery-smartbillions/, 2017. (Accessed 11 April 2021).
[71] L. Brent, A. Jurisevic, M. Kong, et al., Vandal: A scalable security analysis framework for smart contracts, arXiv, 2018 preprint.
[72] A. Roan, How Spankchain Got Hacked. https://medium.com/swlh/how-spankchain-got-hacked-af65b933393c, March 27, 2020. Accessed April 9, 2021.
[73] P. Litke, J. Stewart, BGP Hijacking for Cryptocurrency Profit, Secureworks, 7 August 2014. Available online: https://www.secureworks.com/research/bgp-hijacking-for-cryptocurrency-profit. (Accessed 9 April 2021).
[74] J. Wilcke, The Ethereum Network is Currently Undergoing a DoS Attack, Ethereum Foundation Blog, September 22, 2016. Available online: https://blog.ethereum.org/2016/09/22/ethereum-network-currently-undergoing-dos-attack/. (Accessed 9 April 2021).
[75] Aruba, 10 Blockchain and New Age Security Attacks You Should Know, January 22, 2019. Available online: https://blogs.arubanetworks.com/solutions/10-blockchain-and-new-age-security-attacks-you-should-know/. (Accessed 9 April 2021).
[76] Waqas, The Pirate Bay Caught Running Cryptocurrency Mining Script, HackRead, September 17, 2017. Available online: https://www.hackread.com/the-pirate-bay-caught-running-cryptocurrency-mining-script/. (Accessed 9 April 2021).
[77] K. McCarthy, CBS's Showtime Caught Mining Crypto-coins in Viewers' Web Browsers, The Register, September 25, 2017. Available online: https://www.theregister.com/2017/09/25/showtime_hit_with_coinmining_script/. (Accessed 9 April 2021).
[78] M. Beedham, Hackers secretly ran cryptocurrency mining malware on Indian government sites, 17 Sep 2018. Available online: https://thenextweb.com/news/indian-government-cryptocurrency-coinhive. Accessed: 9 Apr 2021.
[79] T. Claburn, Crypto-jackers Enlist Google Tag Manager to Smuggle Alt-coin Miners, The Register, November 22, 2017. Available online: https://www.theregister.com/2017/11/22/cryptojackers_google_tag_manager_coin_hive/. (Accessed 9 April 2021).
[80] M. Maunder, WordPress Plugin Banned for Crypto Mining, Wordfence, November 8, 2017. Available online: https://www.theregister.com/2017/11/22/cryptojackers_google_tag_manager_coin_hive/. (Accessed 9 April 2021).
[81] T. Mursch, Over 100,000 Drupal Websites Vulnerable to DRUPALGEDDON 2 (CVE-2018-7600), Bad Packets, June 4, 2018. Available online: https://badpackets.net/over-100000-drupal-websites-vulnerable-to-drupalgeddon-2-cve-2018-7600/. (Accessed 9 April 2021).
[82] T. Cantisano, YouTube Ads Hijacked Visitors Computers to Mine Cryptocurrency, Neowin, January 26, 2018. Available online: https://www.neowin.net/news/youtube-ads-hijacked-visitors-computers-to-mine-cryptocurrency/. (Accessed 9 April 2021).
[83] C. Osborne, MikroTik Routers Enslaved in Massive Coinhive Cryptojacking Campaign, ZDNet, August 3, 2018. Available online: https://www.zdnet.com/article/mikrotik-routers-enslaved-in-massive-coinhive-cryptojacking-campaign/. (Accessed 9 April 2021).
[84] L. Kelion, Starbucks Cafe's Wi-Fi Made Computers Mine Crypto-Currency, BBC News, December 13, 2017.
[85] C. Cimpanu, IOTA Cryptocurrency Users Lose $4 Million in Clever Phishing Attack, Bleepingcomputer, 2018. Available online, https://www.bleepingcomputer.com/news/security/iota-cryptocurrency-users-lose-4-million-in-clever-phishing-attack/ (Accessed: 9 Apr 2021).
[86] L. Cuen, IOTA Being Shut Off is the Latest Chapter in an Absurdist History, CoinDesk, February 26, 2020. Available online: https://www.coindesk.com/business/2020/02/25/iota-being-shut-off-is-the-latest-chapter-in-an-absurdist-history/. (Accessed 9 April 2021).
[87] L. Breidenbach, P. Daian, F. Tramer, A. Juels, Enter the Hydra: towards principled bug bounties and exploit-resistant smart contracts, in: 27th USENIX Security Symposium; 15–17 Aug 2018; Baltimore, MD, USA, USENIX Association, Berkeley, CA, USA, 2018, pp. 1335–1352.
[88] E. Hildenbrandt, M. Saxena, N. Rodrigues, et al., KEVM: a complete formal semantics of the ethereum virtual machine, in: 2018 IEEE 31st Computer Security Foundations Symposium (CSF); 9–12 Jul 2018; Oxford, UK, IEEE, Piscataway, NJ, USA, 2018, pp. 204–217.
[89] L. Luu, Oyente: An Analysis Tool for Smart Contracts. https://loiluu.com/oyente.html, 2016. Accessed April 5, 2021.
[90] P. Tsankov, A. Dan, D. Drachsler-Cohen, et al., Securify: practical security analysis of smart contracts, in: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS'18); 15–19 Oct 2018; Toronto, ON, Canada, ACM, New York, NY, USA, 2018, pp. 67–82.
[91] S. Kalra, S. Goel, M. Dhawan, et al., ZEUS: analyzing safety of smart contracts, in: Network and Distributed Systems Security (NDSS) Symposium, San Diego, CA, USA, NDSS, Reston, VA, USA, 2018.
[92] L. Luu, D.-H. Chu, H. Olickel, et al., Making smart contracts smarter, in: The 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS'16); 24–28 Oct 2016; Vienna, Austria, ACM, New York, NY, USA, 2016.
[93] Mythril. https://github.com/ConsenSys/mythril, 2018. Accessed Apr 7, 2021.
[94] J. Frank, C. Aschermann, T. Holz, EthBMC: a bounded model checker for smart contracts, in: 29th USENIX Security Symposium; 12–14 Aug 2020; online, USENIX Association, Berkeley, CA, USA, 2020, pp. 2757–2774.
[95] M. Mossberg, F. Manzano, E. Hennenfent, et al., Manticore: a user-friendly symbolic execution framework for binaries and smart contracts, in: 34th IEEE/ACM International Conference on Automated Software Engineering (ASE); 11–15 Nov 2019; San Diego, CA, USA, IEEE, Piscataway, NJ, USA, 2019, pp. 1186–1189.
[96] I. Nikolic, A. Kolluri, I. Sergey, et al., Finding the greedy, prodigal, and suicidal contracts at scale, in: The 34th Annual Computer Security Applications Conference(ACSAC); 3–7 Dec 2018. San Juan, PR, USA, ACM, New York, NY, USA, 2018, pp. 653–663.
[97] N. Grech, M. Kong, A. Jurisevic, et al., MadMax: surviving out-of-gas conditions in ethereum smart contracts, in: Proceedings of the ACM on Programming Languages 2, 2018, pp. 1–27.
[98] C.F. Torres, J. Schütte, R. State, Osiris: hunting for integer bugs in ethereum smart contracts, in: 34th Annual Computer Security Applications Conference (ACSAC); 3–7 Dec 2018; San Juan, PR, USA, ACM, New York, NY, USA, 2018, pp. 664–676.
[99] M. Zhang, X. Zhang, Y. Zhang, Z. Lin, TxSpector: uncovering attacks in ethereum from transactions, in: 29th USENIX Security Symposium; 12–14 Aug 2020; online, USENIX Association, Berkeley, CA, USA, 2020, pp. 2775–2792.
[100] S. Zhou, Z. Yang, J. Xiang, et al., An ever-evolving game: evaluation of real-world attacks and defenses in ethereum ecosystem, in: 29th USENIX Security Symposium; 12–14 Aug 2020; online, USENIX Association, Berkeley, CA, USA, 2020, pp. 2793–2810.
[101] C.F. Torres, M. Steichen, The art of the scam: demystifying honeypots in ethereum smart contracts, in: 28th USENIX Security Symposium; 14–16 Aug 2019; Santa Clara, CA, USA, USENIX Association, Berkeley, CA, USA, 2019, pp. 1591–1607.
[102] A. Gervais, G.O. Karame, K. Wüst, et al., On the security and performance of proof of work blockchains, in: 2016 ACM SIGSAC Conference on Computer and Communications Security; 24–28 Oct 2016; Vienna, Austria, ACM, New York, NY, USA, 2016, pp. 3–16.
[103] R. Zhang, B. Preneel, Lay down the common metrics: evaluating proof-of-work consensus protocols' security, in: 2019 IEEE Symposium on Security and Privacy; 19–23 May 2019; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2019, pp. 175–192.
[104] B. Jiang, Y. Liu, W. Chan, ContractFuzzer: fuzzing smart contracts for vulnerability detection, in: Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering; 3–7 Sep 2018; Montpellier, France, IEEE, Piscataway, NJ, USA, 2018, pp. 259–269.
[105] C. Liu, H. Liu, Z. Cao, et al., ReGuard: finding reentrancy bugs in smart contracts, in: The 40th International Conference on Software Engineering: Companion; 27 May–3 Jun 2018; Gothenburg, Sweden, IEEE, Piscataway, NJ, USA, 2018, pp. 65–68.
[106] Y. Fu, M. Ren, F. Ma, et al., EVMFuzzer: detect EVM vulnerabilities via fuzz testing, in: 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering; 26–30 Aug 2019. Tallinn, Estonia, ACM, New York, NY, USA, 2019, pp. 1110–1114.
[107] V. Wustholz, M. Christakis, HARVEY: a greybox fuzzer for smart contracts, in: The 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering; 8–13 Nov 2020; Virtual Event, ACM, New York, NY, USA, 2020, pp. 1398–1409.
[108] L. Luu, Y. Velner, J. Teutsch, P. Saxena, SMARTPOOL: practical decentralized pooled mining, in: 26th USENIX Security Symposium, Vancouver, BC, Canada; 16–18 Aug 2017. Vancouver, BC, Canada, USENIX Association, Berkeley, CA, USA, 2017, pp. 1409–1426.
[109] M. Drijvers, K. Edalatnejad, B. Ford, et al., On the security of two-round multi-signatures, in: 40th IEEE Symposium on Security and Privacy; 19–23 May 2019; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2019, pp. 1084–1101.
[110] M. Drijvers, S. Gorbunov, G. Neven, Pixel: multi-signatures for consensus, in: 29th USENIX Security Symposium; 12–14 Aug 2020; online, USENIX Association, Berkeley, CA, USA, 2020, pp. 2093–2110.
[111] Y. Sun, A. Edmundson, N. Feamster, M. Chiang, P. Mittal, Counter-RAPTOR: safeguarding tor against active routing attacks, in: IEEE Symposium on Security and Privacy; 22–26 May 2017; San Jose, CA, USA, IEEE, Piscataway, NJ, USA, 2017, pp. 977–992.
[112] F. Zhang, E. Cecchetti, K. Croman, A. Juels, E. Shi, Town crier: an authenticated data feed for smart contracts, in: 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16); 24–28 Oct 2016; Vienna, Austria, ACM, New York, NY, USA, 2016, pp. 270–282.
[113] A. Mavridou, A. Laszka, Tool demonstration: FSolidM for designing secure ethereum smart contracts, in: International Conference on Principles of Security and Trust; 14–20 Apr 2018; Thessaloniki, Greece, Springer, Cham, Switzerland, 2018, pp. 270–277.
[114] H. Kalodner, S. Goldfeder, X. Chen, et al., Arbitrum: scalable, private smart contracts, in: 27th USENIX Security Symposium; 15–17 Aug 2018; Baltimore, MD, USA, USENIX Association, Berkeley, CA, USA, 2018, pp. 1353–1370.
[115] S. So, M. Lee, J. Park, et al., VERISMART: a highly precise safety verifier for ethereum smart contracts, in: 2020 IEEE Symposium on Security and Privacy; 17–21 May 2020; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2020, pp. 1678–1694.
[116] S. Amani, M. Bégel, M. Bortin, M. Staples, Towards verifying ethereum smart contract bytecode in Isabelle/HOL, in: The 7th ACM SIGPLAN International

Conference on Certified Programs and Proofs; 8–9 Jan 2018; Los Angeles. CA. USA, ACM, New York, NY, USA, 2018, pp. 66–77.
[117] T. Abdellatif, K.-L. Brousmiche, Formal verification of smart contracts based on users and blockchain behaviors models, in: The 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS); 26–28 Feb 2018; Paris, France, IEEE, Piscataway, NJ, USA, 2018, pp. 1–5.
[118] T. Sun, W. Yu, A formal verification framework for security issues of blockchain smart contracts, Electronics 9 (2) (2020), 225.
[119] A. Permenev, D. Dimitrov, P. Tsankov, et al., VerX: safety verification of smart contracts, in: 2020 IEEE Symposium on Security and Privacy; 17–21 May 2020; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2020, pp. 1661–1677.
[120] A. Kosba, A. Miller, E. Shi, Z. Wen, C. Papamanthou, Hawk: the blockchain model of cryptography and privacy-preserving smart contracts, in: 2016 IEEE Symposium on Security and Privacy; 22–26 May 2016; San Jose, CA, USA, IEEE, Piscataway, NJ, USA, 2016, pp. 839–858.
[121] M. Tran, L. Luu, M.S. Kang, I. Bentov, P. Saxena, Obscuro: a bitcoin mixer using trusted execution environments, in: 34th Annual Computer Security Applications Conference (ACSAC); 3–7 Dec 2018; San Juan, PR, USA, ACM, New York, NY, USA, 2018, pp. 692–701.
[122] T. Kerber, A. Kiayias, M. Kohlweiss, V. Zikas, Ouroboros crypsinous: privacy-preserving proof-of-stake, in: 2019 IEEE Symposium on Security and Privacy; 19–23 May 2019; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2019, pp. 157–174.
[123] S. Matetic, K. Wüst, M. Schneider, et al., Bite: bitcoin lightweight client privacy using trusted execution, in: 28th USENIX Security Symposium; 14–16 Aug 2019; Santa Clara, CA, USA, USENIX Association, Berkeley, CA, USA, 2019, pp. 783–800.
[124] S. Bowe, A. Chiesa, M. Green, et al., Zexe: enabling decentralized private computation, in: 2020 IEEE Symposium on Security and Privacy; 18–21 May 2020; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2020, pp. 947–964.
[125] F. Tramèr, D. Boneh, K. Paterson, Remote side-channel attacks on anonymous transactions, in: 29th USENIX Security Symposium; 12–14 Aug 2020; online, USENIX Association, Berkeley, CA, USA, 2020, pp. 2739–2756.
[126] T. Wright, Four-year Anniversary of Bitfinex Hack, and $12M of Stolen BTC Moved, Cointelegraph, Aug 4, 2020. Available online: https://cointelegraph.com/news/four-year-anniversary-of-bitfinex-hack-and-12m-of-stolen-btc-moved. (Accessed 7 April 2021).
[127] Ethereum, Upgrading Ethereum to radical new heights. https://ethereum.org/en/eth2/, March 1, 2021.
[128] S. Dziembowski, L. Eckey, S. Faust, et al., Perun: virtual payment hubs over cryptocurrencies, in: 2019 40th IEEE Symposium on Security and Privacy; 19–23 May 2019; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2019, pp. 106–123.
[129] P. Gazi, A. Kiayias, D. Zindros, Proof-of-Stake sidechains, in: 2019 40th IEEE Symposium on Security and Privacy; 19–23 May 2019; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2019, pp. 139–156.
[130] H. Yu, I. Nikolic, R. Hou, P. Saxena, OHIE: blockchain scaling made simple, in: 2020 41st IEEE Symposium on Security and Privacy; 18–21 May 2020; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2020, pp. 90–105.
[131] Visa, Visa fact sheet. https://usa.visa.com/dam/VCOM/download/corporate/media/visanet-technology/aboutvisafactsheet.pdf, March 1, 2021.
[132] IBM, IBM Blockchain Suppy Chain Solutiuons. https://www.ibm.com/uk-en/blockchain/industries/supply-chain. Accessed July 31, 2021.
[133] VeChain, VeChain Solution Overview. https://vechain.com/solution/logistics. Accessed July 31, 2021.
[134] X. Yu, H. Guo, System, Device and Method for Blockchain-Base Data Exchange, February 5, 2020. Singaporean Patent Application No. 10202000875X.
[135] W. Zou, D. Lo, P.S. Kochhar, et al., Smart Contract Development: Challenges and Opportunities, IEEE Trans. Software Eng., 47 (10) (2019) 2084–2106.
[136] E. Germany, The Crypto SWOT Team Investigates EOS, steemit, 2018. Available online: https://steemit.com/eosgermany/@eosgermany/the-crypto-swot-team-investigates-eos.
[137] P. Daian, S. Goldfeder, T. Kell, et al., Flash Boys 2.0: frontrunning in decentralized exchanges, miner extractable value, and consensus instability, in: 2020 41st IEEE Symposium on Security and Privacy(SP); 18–21 May 2020; San Francisco, CA, USA, IEEE, Piscataway, NJ, USA, 2020, pp. 910–927.
[138] Y. Zhou, D. Kumar, S. Bakshi, et al., Erays: reverse engineering Ethereum's opaque smart contracts, in: 27th USENIX Security Symposium; 15–17 Aug 2018; Baltimore, MD, USA, USENIX Association, Berkeley, CA, USA, 2018, pp. 1371–1385.
[139] H. Kalodner, M. Möser, K. Lee, et al., BlockSci: design and applications of a blockchain analysis platform, in: 29th USENIX Security Symposium; 12–14 Aug 2020; online, USENIX Association, Berkeley, CA, USA, 2020, pp. 2721–2738.
[140] G. Xu, B. Guo, C. Su, X. Zheng, K. Liang, D.S. Wong, H. Wang, Am I eclipsed? A smart detector of eclipse attacks for ethereum, Comput. Secur. 88 (2020).
[141] P.W. Shor, Algorithms for quantum computation: discrete logarithms and factoring, in: 35th Annual Symposium on Foundations of Computer Science; 20–22 Nov 1994 ; Santa Fe, NM, USA, IEEE, Piscataway, NJ, USA, 1994, pp. 124–134.
[142] A. Bouguera, How will quantum computing affect blockchain? Consensys (December 3, 2019). Available online: https://consensys.net/blog/developers/how-will-quantum-supremacy-affect-blockchain/. (Accessed 31 July 2021).
[143] NIST's Post-Quantum Cryptography Program Enters 'Selection Round', National Institute of Standards and Technology (NIST), July 22, 2020. Available online: https://www.nist.gov/news-events/news/2020/07/nists-post-quantum-cryptography-program-enters-selection-round. (Accessed 31 July 2021).
[144] I. Barmes, B. Bosch, Quantum Computers and the Bitcoin Blockchain, Deloitte, March 13, 2021. Available online: https://www.nist.gov/news-events/news/2020/07/nists-post-quantum-cryptography-program-enters-selection-round. (Accessed 31 July 2021).
[145] L. M, IOTA Price Prediction 2021 and Beyond: What to Expect? BigDegree, January 25, 2021. Available online: https://www.bitdegree.org/crypto/tutorials/iota-price-prediction.
[146] S. Sayadi, S.B. Rejeb, Z. Choukair, Blockchain challenges and security schemes: a survey, in: Seventh International Conference on Communications and Networking (ComNet); 1–3 Nov 2018; Hammamet, Tunisia, IEEE, Piscataway, NJ, USA, 2018, pp. 1–7.
[147] M. Saad, J. Spaulding, L. Njilla, et al., Exploring the attack surface of blockchain: a systematic overview, IEEE Communications Surveys & Tutorials 22 (3) (2020) 1977–2008.
[148] D. Dasgupta, J.M. Shrein, K.D. Gupta, A survey of blockchain from security perspective, Journal of Banking and Financial Technology 3 (2019) 1–17.
[149] J. Leng, M. Zhou, J.L. Zhao, Y. Huang, Y. Bian, Blockchain security: a survey of techniques and research directions, IEEE Transactions on Services Computing, 2020.
[150] W. Chen, Z. Xu, S. Shi, et al., A survey of blockchain applications in different domains, in: International Conference on Blockchain Technology and Applications (ICBTA); 10–12 Dec 2018; Xi'an, China, ACM, New York, NY, USA, 2018, pp. 17–21.