Decentralized platforms: Goals, challenges, and solutions

Sergii Grybniak
Institute of Computer Systems
Odesa Polytechnic State University
Odesa, Ukraine
s.s.grybniak@op.edu.ua

Yevhen Leonchyk
Faculty of Mathematics, Physics, and Information Technologies
Odesa I.I. Mechnikov National University
Odesa, Ukraine
leonchyk@onu.edu.ua

Ruslan Masalskyi
Faculty of Mathematics, Physics, and Information Technologies
Odesa I.I. Mechnikov National University
Odesa, Ukraine
masalskyi@stud.onu.edu.ua

Igor Mazurok
Faculty of Mathematics, Physics, and Information Technologies
Odesa I.I. Mechnikov National University
Odesa, Ukraine
mazurok@onu.edu.ua

Oleksandr Nashyvan
Institute of Computer Systems
Odesa Polytechnic State University
Odesa, Ukraine
o.nashyvan@op.edu.ua

Ruslan Shanin
Faculty of Mathematics, Physics, and Information Technologies
Odesa I.I. Mechnikov National University
Odesa, Ukraine
ruslanshanin@onu.edu.ua

***Abstract*— Today there is a trend to deploy both commercial and social applications and services on decentralized platforms. This corresponds to the demands of modern society for openness and transparency of information, freedom of access, and equal rights of participants. However, the more sophisticated architecture of decentralized platforms poses new challenges to the developers of such systems. This paper discusses the goals, tasks, and problems that arise during the transition from traditional (centralized) technologies to decentralized ones, and some potential ways to solve them.**

Keywords—decentralized system, blockchain, consensus protocol, directed acyclic graph, distributed ledger technology, peer-to-peer

I. HISTORICAL INTRODUCTION

Peer-to-peer networks (p2p) are characterized by a distributed architecture in which there is no single administrative server to manage the entire system [1]. In such networks, all participants have the same capabilities, combining both client and server functionality. The main advantage of p2p networks is that even a very large number of faulty nodes will not shut down the system. It can be said that a user community itself controls and are responsible for network performance (in terms of average throughput, latency, resource consumption, transaction size and cost, fault-tolerance, etc), working together toward a common goal. However, the need to negotiate between multiple peers makes decentralized solutions more complex and sophisticated than centralized ones.

The term "peer-to-peer" was first used in the design of the advanced networking architecture by IBM [2], although ideas of this kind had been discussed long before that (e.g. [3]). Its implementation was based on a number of technologies developed earlier, such as Merkle trees [4] and Vault systems [5]. In the 1990s, p2p networks became widespread due to file sharing and illegal file sharing [6]. Now they are also used in diverse financial services including payment systems [7, 8], medical [9] and real estate [10] sectors, support for the Internet of Things (IoT) [11, 12], logistics management [13], energy industry [14, 15], identity document (ID) services [16], e-voting [17], etc. Also, mechanisms ensuring the functioning of p2p networks, such as distributed hash table [18] and BitTorrent [19] technologies, are being improved.

Currently, the concepts of decentralization and blockchain are often used together. Blockchain technology is a database that operates in a decentralized manner and stores chains of blocks of information about users and transactions. The blocks are cryptographically interlinked and stored on the devices of each participant in the network. Blockchain was made possible by the further development of p2p solutions, such as timestamping digital documents [20], decentralized digital currency [21], and cryptographic security of chains [22]. In addition, new economic approaches were needed to incentivize developers, investors, maintainers, and other network participants [23]. A detailed overview of the origins of blockchain technologies can be found, e.g., in [24], [25].

The first implemented blockchain p2p system is the well-known Bitcoin [26], based on Proof-of-Work (PoW) consensus, where network nodes, aka miners, must solve complex cryptographic problems to produce blocks. However, Bitcoin's functionality does not fully meet all of today's requirements. Ethereum is an example of a platform that gives users more capabilities [27]. Its key feature is support for smart contracts, which allows the platform to be considered an Ethereum Virtual Machine (EVM). This has given a new direction to many online industries, from gaming to the financial industry. At the same time, this was not sufficient for many enterprise-class applications that also demand high performance and scalability. Subsequently, alternative platforms appeared, aimed at fixing these limitations. At this point, we cannot say that these issues have acceptable solutions; their active discussion and development of new approaches continues.

Experience in use has shown that PoW-based systems eventually reach a significant level of power consumption, as

https://doi.org/10.1109/RTSI55261.2022.9905225

the total computing power of network security equipment increases [28]. High power consumption does not meet the demands of modern society for environmentally friendly solutions that conserve energy. Moreover, as hired workers, participants of the PoW protocol have little interest in the further development of the network. Both of these drawbacks have been eliminated by the concept of Proof-of-Stake (PoS), in which participants are essentially shareholders and the probability to produce a block depends on their stake in a deterministic (pseudo-random) way [29]. Today, this approach to the development of decentralized systems is the most developed. To increase the level of participants' involvement and interest, in addition to personal funds, their activity and reputation rates are also used as incentives [30].

The development of Distributed Ledger Technology (DLT) is not limited to blockchain. Recently, a new architecture paradigm – Directed Acyclic Graph (DAG) – has become increasingly popular. Systems based on this approach may lend themselves to the evolution of next-generation blockchain technology due to their high scalability [31, 32]. However, the representation of DLT structure as a DAG poses new challenges, which leading decentralized systems solve in different ways, modifying existing protocols and creating new ones [33].

II. Definitions and Requirements

Decentralization in modern computing systems is generally understood to be the independent execution of all system functions by a significant number of independent servers (nodes), usually with the prior coordination of input data and with the subsequent coordination of the results of work [34]. Thus, there is a responsible execution of all functions under conditions of complete distrust. The presence of hacked, corrupted, or malicious nodes is not merely considered a possibility, but is assumed to be a reliable fact.

It is generally accepted that computer programs and applications run within a platform that provides the necessary computing power, resources and infrastructure. Technically, applications can run either on a single computer or in a distributed manner on several interacting computing systems. ***Decentralized applications*** (dapps) are characterized by running and executing on multiple nodes [35-37]. In effect, it means that a program is executed by several computers that independently come up with results, which are then reconciled. The term is also used in a weaker sense for traditional applications that use data and/or store their results in decentralized information stores like blockchain.

Decentralized data repositories have much in common with those of distributed databases, where a large information redundancy factor is maintained, allowing data integrity to be restored when data is lost or distorted on a large portion of the servers. Potentially, the concept of decentralized repositories allows the joint work of nodes to isolate and restore the preserved correct chain of data from any number of corrupted ones. Typically, decentralized storage does not support – and explicitly prohibits – data correction or deletion [38, 39].

We understand a ***decentralized platform*** to be a distributed computer system that provides a complete infrastructure to support the functioning of decentralized applications and services. As a rule, such a platform provides the means for:

- decentralized information storage and query processing;
- decentralized execution of program code;
- communication between users and applications in any combination;
- decentralized economic support for commercial and financial interactions.

The technical basis of a decentralized platform is a set of nodes interacting on certain protocols [1, 40, 41]. Nodes can be functionally divided into several groups, but within the current time of functioning of the network, the nodes of the group are fully equal and the number of nodes in the groups is significant. At certain tacts of protocol execution, a node can play a special or even a crucial role. However, at different tacts such a node changes so that, in general over a long period of time, all nodes in the group have the opportunity to make a relatively equal impact on the functioning of the system. This requirement means that at any given time, a certain node can be a decision point (e.g., to create information blocks for recording in a shared repository – a distributed ledger [42-45]). Thus, nodes are functionally equivalent within the role group of the protocol.

III. Creation Goals

Typically, decentralized solutions are inferior to centralized ones in terms of labor intensity and efficiency. However, there are several important reasons for using decentralized technologies. First of all, building global decentralized computer systems is underscored by the desire to avoid outside influence on the functioning of the system:

- political and governmental mechanisms;
- economic factors of regional nature (increased cost of electricity, communications, qualified service personnel, etc.);
- damage to equipment or communication channels due to natural causes, or as a result of an attack;
- malicious attacks to gain control of the system or to tamper with data.

Decentralized systems support the functioning of the interaction system in the absence of mutual trust between all participants. The technology itself implies openness (transparency) and verifiability of data and executable code in real time. In addition, decentralized solutions, as well as edge computing technologies, can improve the delivery of content and services by placing access points "closer" to users or globally distributed resources.

IV. Challenges

To achieve each of the above goals, a different set of tasks must be accomplished. The common set necessarily includes the following issues.

Decentralization. There are no nodes or groups with special powers that cannot be accessed by a normal user.

Augmented Data. There is a constant challenge of adding new data to a shared distributed ledger. A decentralized

platform within the protocol determines who, when, and what data can be written [46].

Ordering. For stored data in many real-world applications (especially financial applications), the order in which data are added to the shared ledger is important, because without it, the business logic of most applications does not work [8].

Data Synchronization. All honest (properly operating and following the network protocol) nodes should have the same state of the ledger. Given that information on the network cannot propagate instantaneously, this requirement is formulated as consistency. In other words, for two honest nodes, the ledger of one is always the prefix of the ledger of the other [47].

Finalization. In most systems, a situation arises where different nodes assume different versions of adding data to the ledger. The problem of finalization in weak form consists in the probabilistic or deterministic definition of a common prefix expanding with time of all versions of the ledger in honest nodes [48]. In the strong form (fast finalization), the problem is to create a protocol with an unambiguous and indisputable fixation of the order and content of new data before it is fully distributed over the network [49].

Forgery. As a result of malicious intent or operational errors, the data in the ledger may become corrupted on parts of the nodes. Thus, it is necessary to have a procedure for determining the correct version of the data, even if the data has survived on just one node [50].

Availability. Any query to a distributed system culminates with a correct response, but there is no guarantee that the responses of all nodes in the system will be the same [51]. Even honest nodes are often in different states, primarily because of the different state of the yet-unfinalized tail of the ledger. However, there are systems, such as those based on independent shards [52], in which the decentralized system operates in a state of permanent network separation at the level of the ledger.

Scalability. The system must have mechanisms for adapting to changes in the flow of transactions within very wide limits [53-56]. Scalability is sometimes understood as a response to an increase in the number of network nodes. However, this number is usually automatically adjusted in an economic way, so that there are enough nodes to safely perform the necessary functions [57]. Thus, an increase in the number of nodes as opposed to an increase in transaction flow is not unconditionally positive for system development.

Security. The system as a whole must be resilient to failures and attacks, being able to recover from all honest nodes [58-61].

Entrance. To enter a decentralized system, a new node or user needs to establish a connection to some nodes in the network using a certain protocol. This usually requires knowing their ip-address and port. Obtaining this information takes place outside the network – fixed addresses of several bootstrap nodes built into the node code, publishing addresses on websites, transmitting via e-mail or messengers – which are usually described in the technical documentation.

V. SOLUTIONS AND TECHNOLOGIES

Blockchain in the narrow sense is a sequence of transaction blocks in which each successive block contains the hash code of the previous block [24, 25]. Such a structure makes it difficult to forge the past – if changes are made to any block, changes must be made to all subsequent blocks. Some consensus protocols (e.g., PoW) further complicate such tampering by requiring that the value of some irreversible function from its content be picked up and included in the block. The creation of blocks is done by nodes that constantly monitor the emergence of new blocks. This allows for the introduction of requirements of reference to the last (newest) known block. In the finalization process, side chains are discarded and all blocks are lined up in one strict sequence.

Directed Acyclic Graph (DAG). Unlike blockchain technology with linear sequenced blocks, in such systems, blocks/transactions form referential structures known in graph theory as DAG since they can (and sometimes must) reference several previous blocks/transactions [31-33]. In case of blockless systems, transactions are created directly by network participants rather than by special nodes (miners, blocksigners, etc.). Some systems impose an additional restriction on references, allowing, as in blockchain, only the last known blocks/transactions to be referenced.

In all DAG-based systems there is a problem of linearization (ordering), because the referential structure of DAG in the general case defines a partially ordered set. In such cases, it is necessary to achieve a resolution of block order uncertainty by some additional actions. In addition, DAG block systems are characterized by the problem of repeated inclusion of the same transaction in blocks.

Conflict-free replicated data type (CRDT) is an alternative to consensus-based technologies and leader opinion replication [62]. CRDT is a data type with a conflict resolution algorithm that allows each of the network nodes to update the ledger independently, without consulting with other nodes. It is assumed that information about the changes made will be distributed over the network (e.g. via the Gossip protocol) and cause a wave of updates. In the process of change propagation, the state of the ledger gradually stabilizes relative to each particular transaction. CRDT-based decentralized systems can propagate either transactions (CmRDT) or the ledger state (CvRDT) across the network. For the system to be practically implemented by CRDT, operations for ledger modification or state merge should have a certain set of properties (e.g. commutativity, associativity, or idempotency).

Content-addressable storage (CAS). To access data in conventional storage, as a rule, index tables are used, which link the value of a primary key that uniquely identifies a data block with its physical storage location. CAS systems specify the means of key construction (as a hash block) and how to determine its physical storage location [63].

Distributed Hash Table (DHT) is an extension of the CAS approach to the case of a peer-to-peer decentralized system with support for p2p connections [18, 64, 65]. Many interesting solutions in this area have been made during the

development of the InterPlanetary File System (IPFS) [66, 67]. We should also pay attention to the implementation of DHT in the Kademlia project [68].

Digital Signature. Asymmetric encryption algorithms, often based on elliptic cryptography, such as signature aggregation based on the Boneh-Lynn-Shacham (BLS) method, are used to identify and authenticate nodes and users [69, 70]. Quantum-safe algorithms based on non-interactive proofs with zero-disclosure, such as [71, 72], are currently being developed.

Consensus protocols. Consensus decisions in some cases require that all honest participants arrive at the same decision. Consensus protocols solve this problem [30, 48, 73-75]. Such protocols can be deterministic (guaranteed to lead to success) and probabilistic. According to the conditions of applicability, it is customary to distinguish consensuses for private and public networks. Consensus for private networks operates under additional constraints – the protocol participants have a complete list (or at least know the number) of nodes, and the number of foul nodes is limited to some number or a certain fraction of their total number.

For private systems, there are a sufficient number of robust consensus protocols, such as Raft [76] or various modifications of Byzantine Fault Tolerance (BFT) [77, 78]. There are a significant number of other consensus protocols and their modifications with their strengths and weaknesses [30, 75]. The notion of using the PoS approach to store useful information and PoW to perform some socially useful computation looks promising.

Gossip protocols. The Gossip family of protocols solves the problem of distributing information over p2p connections when a complete list of network node addresses is not available [79], which is typical for public networks. For information dissemination, each node, having received a message, transmits it to those nodes whose addresses it knows. At the stage of establishing communication, steps are taken to limit the re-acquisition of data. For the reliability and speed of the protocol, the connectivity of the link graph of the nodes and its diameter is important. However, since the link graph is dynamically changing and completely unknown to any of the participants, guaranteed delivery is generally not possible. In turn, solving the connectivity problem by allocating a certain number of well-known bootstrap nodes leads to some centralization. The acceleration of the work is done by path reduction. To do this, the transferred data are supplemented with the addresses of the transmitting nodes, but this does not contribute to overall safety.

Sharding is a well-established technique in database management systems of splitting (vertical and horizontal sharding) a ledger to solve the problem of scaling. In decentralized systems, sharding involves partitioning a set of nodes into groups with or without appropriate ledger sharding [80, 81]. If only one set of nodes is partitioned, this is done to speed up consensus and reduce the amount of associated communication. If sharding of nodes is performed simultaneously with sharding of the ledger (each shard leads its own ledger), then with significant reduction of network load we get the problem of shard coordinating [82], which can be solved in a synchronous or asynchronous way.

For developing various dapps, from our perspective, the optimal solution is a scalable EVM-based smart contract platform with reasonably fast finality PoS consensus and the programming language Solidity [83]. To facilitate scalability, which is one of the main challenges of decentralized technologies, DAG-based ledger structure and sharding technologies could be applied for achieving high transaction throughput via parallelized block production. In addition, the embedded token approach facilitates dapp developers in producing new applications, as well as effortlessly transferring those already created onto the low-cost maintenance platform. The implemented multiple-tier node system enables the joining of diverse devices into the network, making it widely accessible, and positively influencing the system's decentralization characteristics. Based on this, the Waterfall platform [84] is designed to provide a favorable environment for the provision and consumption of a wide spectrum of enterprise-class services, for conducting business and social activities in a convenient format, within the framework of a public p2p network.

VI. CONCLUSIONS

Decentralized platforms allow for flexible and scalable systems that are resistant to technical failures. Eliminating intermediaries reduces additional costs and eliminates bureaucracy in user interaction. The use of shared resources becomes transparent and secure, even in systems lacking mutual trust. Small and medium-sized businesses are given an easy entry threshold into markets and a level playing field for competition.

On the other hand, the development and deployment of such systems is more complex, and their operation may be accompanied by higher maintenance costs and unstable performance, especially in cases of improper optimization. Regulatory and legal risks may arise, as existing legislation focuses on traditional solutions in which there is a single responsible entity.

The distribution and implementation of decentralized systems has no alternative because centralized solutions do not allow any of the above objectives to be fully realized. If there is a requirement in the task definition to significantly reduce the external influence on the business logic of the system and/or the system's functioning in the absence of mutual trust, decentralization is a reasonable and meaningful approach. There is growing demand for such an approach in a computerized society where information is one of its basic resources.

REFERENCES

[1] R. Schollmeier, I. Gruber, and F. Niethammer, "Protocol for peer-to-peer networking in mobile environments," 12th International Conference on Computer Communications and Networks (IEEE Cat. "BlockNo. 03EX712), IEEE, 2003, pp. 121-127.

[2] P. E. Green, R. J. Chappuis, J. D. Fisher, P. S. Frosch, and C. E. Wood, "A perspective on Advanced Peer-to-Peer Networking," in IBM Systems Journal, 26 (4), 1987, pp. 414-428.

[3] S. Crocker, "Host software," Rfc1., 1969.

[4] R. C. Merkle, “Secrecy, authentication, and public key systems,” Stanford university, 1979.

[5] D. L. Chaum, "Computer Systems Established, Maintained and Trusted by Mutually Suspicious Groups," Electronics Res. Lab., University of California, Berkeley, 1979.

[6] M. Green, "Napster opens Pandora's Box: Examining how file-sharing services threaten the enforcement of copyright on the internet," Ohio St. LJ, 63 (799), 2002.

[7] R. Selkis, “A Messari report: Crypto Theses for 2022,” 2022.

[8] S. M. Werner, D. Perez, L. Gudgeon, A. Klages-Mundt, D. Harz, and W. J. Knottenbelt, “Sok: Decentralized finance (defi),” preprint arXiv:2101.08778, 2021.

[9] J. Kabyemela, "The IOTA Tangle for Electronic Medical Records Systems," 2019.

[10] R. M. Garcia-Teruel, “Legal challenges and opportunities of blockchain technology in the real estate sector,” Journal of Property, Planning and Environmental Law, 2020.

[11] M. Conoscenti, A. Vetro, and J. C. De Martin, “Blockchain for the Internet of Things: A systematic literature review,” IIEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), 2016, pp. 1-6.

[12] X. Wang, X. Zha, W. Ni, R. P. Liu, Y. J. Guo, X. Niu, and K. Zheng, “Survey on blockchain for Internet of Things,” Computer Communications, 136, 2019, pp. 10-29.

[13] M. Pournader, Y. Shi, S. Seuring, and S. L. Koh, “Blockchain applications in supply chains, transport and logistics: a systematic review of the literature,” International Journal of Production Research, 58(7), 2020, pp. 2063-2081.

[14] M. Andoni et al., "Blockchain technology in the energy sector: A systematic review of challenges and opportunities," Renewable and Sustainable Energy Reviews, vol. 100, pp. 143-174, 2019.

[15] Q. Wang and M. Su, “Integrating blockchain technology into the energy sector – from theory of blockchain to research and application of energy blockchain,“ Computer Science Review, 37, 2020.

[16] J. H. Lee, “BIDaaS: Blockchain based ID as a service,” IEEE Access, 6, 2017, pp. 2274-2278.

[17] N. Kshetri and J. Voas, “Blockchain-enabled e-voting,” IEEE Software, 35(4), 2018, pp. 95-99.

[18] M. Harren, J. M. Hellerstein, R. Huebsch, B. T. Loo, S. Shenker, and I. Stoica, “Complex queries in DHT-based peer-to-peer networks,” International Workshop on Peer-to-Peer Systems, Springer, Berlin, Heidelberg, 2002, pp. 242-250.

[19] J. Pouwelse, P. Garbacki, D. Epema, and H. Sips, “The bittorrent p2p file-sharing system: Measurements and analysis,” International workshop on peer-to-peer systems, Springer, Berlin, Heidelberg, 2005, pp. 205-216.

[20] S. Haber and W. S. Stornetta, “How to time-stamp a digital document,” Conference on the Theory and Application of Cryptography, Springer, Berlin, Heidelberg, 1990, pp. 437-455.

[21] N. Szabo “Secure property titles with owner authority,” 1998.

[22] S. Konst, "Secure log files based on cryptographically concatenated entries," Technische Universitat Braunschweig, 2000.

[23] S. Au, Th. Power, “Tokenomics: The Crypto Shift of Blockchains, ICOs, and Tokens,” Packt Publishing Ltd, 2018.

[24] A. T. Sherman, F. Javani, H. Zhang, and E. Golaszewski, “On the origins and variations of blockchain technologies,” IEEE Security & Privacy, 17(1), 2019, pp. 72-77.

[25] M. N. M. Bhutta et al., "A Survey on Blockchain Technology: Evolution, Architecture and Security," in IEEE Access, vol. 9, 2021, pp. 61048-61073.

[26] S. Nakamoto, “Bitcoin: A Peer-to-Peer Electronic Cash System,” 2009.

[27] V. Buterin “A next-generation smart contract and decentralized application platform,” White Paper, 2014.

[28] A. N. Q. Huynh, D. Duong, T. Burggraf, H. T. T. Luong, and N. H. Bui, “Energy consumption and Bitcoin market,” Asia-Pacific Financial Markets, 29(1), 2022, 79-93 pp.

[29] C. T. Nguyen, D. T. Hoang, D. N. Nguyen, D. Niyato, H. T. Nguyen, and E. Dutkiewicz, “Proof-of-stake consensus mechanisms for future blockchain networks: fundamentals, applications and opportunities,” IEEE Access, 7, 2019, pp. 85727-85745.

[30] S. Bano, A. Sonnino, M. Al-Bassam, S. Azouvi, P. McCorry, S. Meiklejohn, and G. Danezis, “SoK: Consensus in the age of blockchains,” Proceedings of the 1st ACM Conference on Advances in Financial Technologies, 2019, pp. 183-198.

[31] F. M. Benčić and I. Podnar Žarko, "Distributed Ledger Technology: Blockchain Compared to Directed Acyclic Graph," IEEE 38th International Conference on Distributed Computing Systems, 2018, pp. 1569-1570.

[32] H. Pervez, M. Muneeb, M. U. Irfan and I. U. Haq, "A Comparative Analysis of DAG-Based Blockchain Architectures," 2018 12th International Conference on Open Source Systems and Technologies (ICOSST), 2018, pp. 27-34.

[33] Q. Wang, J. Yu, S. Chen, and Y. Xiang, “SoK: Diving into DAG-based blockchain systems,” preprint arXiv:2012.06128, 2020.

[34] Y. Patt and S. Patel, “Introduction to computing systems,” McGraw-Hill, 2003.

[35] W. Cai, Z. Wang, J. B. Ernst, Z. Hong, C. Feng, and V. C. Leung, “Decentralized applications: The blockchain-empowered software system,” IEEE Access, 2018, pp. 53019-53033.

[36] S. Raval, “Decentralized applications: harnessing Bitcoin's blockchain technology,” O'Reilly Media, Inc., 2016.

[37] C. Antal, T. Cioara, I. Anghel, M. Antal, and I. Salomie, “Distributed ledger technology review and decentralized applications development guidelines,” Future Internet, 2021, p. 62.

[38] A. Lakshman, and P. Malik, “Cassandra: a decentralized structured storage system,” ACM SIGOPS Operating Systems Review, 2010, pp. 35-40.

[39] D. Vorick and L. Champine, “Sia: Simple decentralized storage,” Retrieved May, 2014, p. 2018.

[40] D. Andriesse and H. Bos, “An analysis of the zeus peer-to-peer protocol,” Technical Report IR-CS-74, 2014.

[41] I. Wang, and I. Taylor, “P2PS (peer-to-peer simplified),” Proceedings of 13th Annual Mardi Gras Conference-Frontiers of Grid Applications and Technologies, 2005, pp. 54-59.

[42] R. Maull, P. Godsiff, C. Mulligan, A. Brown, and B. Kewell, “Distributed ledger technology: Applications and implications,” Strategic Change, 2017, pp. 481-489.

[43] M. Rauchs, A. Glidden, B. Gordon, G. C. Pieters, M. Recanatini, F. Rostand, and B. Z. Zhang, “Distributed ledger technology systems: A conceptual framework,” 2018.

[44] N. El Ioini and C. Pahl, “A review of distributed ledger technologies,” OTM Confederated International Conferences "On the Move to Meaningful Internet Systems," Springer, Cham, 2018, pp. 277-288.

[45] D. Burkhardt, M. Werling, and H. Lasi, “Distributed ledger,” IEEE international conference on engineering, technology and innovation (ICE/ITMC), IEEE, 2018, pp. 1-9.

[46] Z. He, L. Xie, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, “Data augmentation revisited: Rethinking the distribution gap between clean and augmented data,” preprint arXiv:1909.09148, 2019.

[47] S. Biswas, K. Sharif, F. Li, Z. Latif, S. S. Kanhere, and S. P. Mohanty, “Interoperability and synchronization management of blockchain-based decentralized e-health systems,” IEEE Transactions on Engineering Management, 2020, pp. 1363-1376.

[48] Y. Xiao, N. Zhang, W. Lou, and Y. T. Hou, “A survey of distributed consensus protocols for blockchain networks,” IEEE Communications Surveys & Tutorials, 2020, pp. 1432-1465.

[49] Y. Pan, B. Pi, and J. Sun, “Plume: Fast Finality Blockchain without Single Failure Point,” 2nd Asia Service Sciences and Software Engineering Conference, 2021, pp. 18-27.

[50] W. Liang, Y. Fan, K. C. Li, D. Zhang, and J. L. Gaudiot, “Secure data storage and recovery in industrial blockchain network environments,” IEEE Transactions on Industrial Informatics, 2020, pp. 6543-6552.

[51] I. Weber, V. Gramoli, A. Ponomarev, M. Staples, R. Holz, A. B. Tran, and P. Rimba, "On availability for blockchain-based systems," IEEE 36th Symposium on Reliable Distributed Systems (SRDS), IEEE, 2017, pp. 64-73.

[52] M. J. Amiri, D. Agrawal, and A. El Abbadi, "On sharding permissioned blockchains," IEEE International Conference on Blockchain (Blockchain), IEEE, 2019, pp. 282-285.

[53] Q. Zhou, H. Huang, Z. Zheng, and J. Bian, "Solutions to scalability of blockchain: A survey," IEEE Access, 2020, pp. 16440-16455.

[54] A. Chauhan, O. P. Malviya, M. Verma, and T. S. Mor, "Blockchain and scalability," 2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C). IEEE, 2018, pp. 122-128.

[55] S. Kim, Y. Kwon, and S. Cho, "A survey of scalability solutions on blockchain," International Conference on Information and Communication Technology Convergence (ICTC), IEEE, 2018, pp. 1204-1207.

[56] J. Xie, F. R. Yu, T. Huang, R. Xie, J. Liu, and Y. Liu, "A survey on the scalability of blockchain systems," IEEE Network, 2019, pp. 166-173.

[57] K. Lau, "Ethereum 2.0. An Introduction," 2020.

[58] R. Zhang, R. Xue, and L. Liu, "Security and privacy on blockchain," ACM Computing Surveys (CSUR), 2019, pp. 1-34.

[59] I. C. Lin, and T. C. Liao, "A survey of blockchain security issues and challenges," Int. J. Netw. Secur, 2017, pp. 653-659.

[60] X. Li, P. Jiang, T. Chen, X. Luo, and Q. Wen, "A survey on the security of blockchain systems," Future Generation Computer Systems, 2020, pp. 841-853.

[61] D. Dasgupta, J. M. Shrein, and K. D. Gupta, "A survey of blockchain from security perspective," Journal of Banking and Financial Technology, 2019, pp. 1-17.

[62] M. Shapiro, N. Preguiça, C. Baquero, and M. Zawirski, "Conflict-free replicated data types," Symposium on Self-Stabilizing Systems, Springer, Berlin, Heidelberg, 2011, pp. 386-400.

[63] N. Tolia, M. Kozuch, M. Satyanarayanan, B. Karp, T. C. Bressoud, and A. Perrig, "Opportunistic Use of Content Addressable Storage for Distributed File Systems," USENIX Annual Technical Conference, General Track, 2003, pp. 127-140.

[64] F. F. E. Dabek, "A distributed hash table," Massachusetts Institute of Technology, 2005.

[65] M. F. Kaashoek and D. R. Karger, "Koorde: A simple degree-optimal distributed hash table," International Workshop on Peer-to-Peer Systems, Springer, Berlin, Heidelberg, 2003, pp. 98-107.

[66] S. Muralidharan and H. Ko, "An InterPlanetary file system (IPFS) based IoT framework," IEEE international conference on consumer electronics (ICCE), IEEE, 2019, pp. 1-2.

[67] M. Nazet et al., "A secure data sharing platform using blockchain and interplanetary file system," Sustainability, 11 (24), 2019.

[68] P. Maymounkov and D.Mazieres, "Kademlia: A peer-to-peer information system based on the xor metric," International Workshop on Peer-to-Peer Systems, Springer, Berlin, Heidelberg, 2002, pp. 53-65.

[69] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps," International conference on the theory and applications of cryptographic techniques, Springer, Berlin, Heidelberg, 2003, pp. 416-432.

[70] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," International conference on the theory and application of cryptology and information security, Springer, Berlin, Heidelberg, 2001, pp. 514-532.

[71] M. R. Albrecht et al., "Algebraic cryptanalysis of STARK-friendly designs: application to MARVELlous and MiMC," International Conference on the Theory and Application of Cryptology and Information Security, Springer, Cham, 2019, pp. 371-397.

[72] T. Ashur and S. Dhooghe, "MARVELlous: a STARK-friendly family of cryptographic primitives," Cryptology ePrint Archive, 2018.

[73] S. M. H. Bamakan, A. Motavali, and A. B. Bondarti, "A survey of blockchain consensus algorithms performance evaluation criteria," Expert Systems with Applications, 2020.

[74] C. Zhang, C. Wu, and X. Wang, "Overview of Blockchain consensus mechanism," Proceedings of the 2nd International Conference on Big Data Engineering, 2020, pp. 7-12.

[75] Q. Wang, J. Huang, S. Wang, Y. Chen, P. Zhang, and L. He, (2020). "A comparative study of blockchain consensus algorithms," Journal of Physics: Conference Series, vol. 1437 (1), IOP Publishing, 2020.

[76] H. Howard, "ARC: Analysis of Raft Consensus," University of Cambridge, Computer Laboratory, UCAM-CL-TR-857, 2014.

[77] L. Lamport and M. Fischer, "Byzantine generals and transaction commit protocols," 1982.

[78] M. Castro and B. Liskov, "Practical byzantine fault tolerance," OsDI, 1999, pp 173-186.

[79] K. Birman, "The promise, and limitations, of gossip protocols," ACM SIGOPS Operating Systems Review, 2007, pp. 8-13.

[80] M. Zamani, M. Movahedi, and M. Raykova, "Rapidchain: Scaling blockchain via full sharding," Proceedings of ACM SIGSAC conference on computer and communications security, 2018, pp. 931-948.

[81] H. Dang, T. T. A. Dinh, D. Loghin, E. C. Chang, Q. Lin, and B. C. Ooi, "Towards scaling blockchain systems via sharding," Proceedings of the international conference on management of data, 2019, pp. 123-140.

[82] Y. Liu, J. Liu, J. Yin, G. Li, H. Yu, and Q. Wu, "Cross-shard transaction processing in sharding blockchains," International Conference on Algorithms and Architectures for Parallel Processing, Springer, Cham, 2020, pp. 324-339.

[83] C. Dannen, "Introducing Ethereum and solidity," Berkeley: Apress, 1, 2017.

[84] S. Grybniak, D. Dmytryshyn, Y. Leonchyk, I. Mazurok, O. Nashyvan, and R. Shanin, "Waterfall: A Scalable Distributed Ledger Consensus Protocol," unpublished.