Preprint: Hanna Grodzicka, Michal Kedziora, and Lech Madeyski. "Security and Scalability in Private Permissionless Blockchain: Problems and Solutions Leading to Creating Consent-as-a-Service (CaaS) Deployment," in Advances in Computational Collective Intelligence (K. Wojtkiewicz, J. Treur, E. Pimenidis, and M. Maleszka, eds.), (Cham), pp. 278–289, Springer International Publishing, 2021. DOI: 10.1007/978-3-030-88113-9_22

Preprint: https://madeyski.e-informatyka.pl/download/GrodzickaEtAl21.pdf

Security and Scalability in Private Permissionless Blockchain: Problems and Solutions Leading to Creating Consent-as-a-Service (CaaS) Deployment

Hanna Grodzicka[1 [0000−0002−0142−3270]], Michal Kedziora[1 [0000−0002−7764−1303]], and Lech Madeyski[1 [0000−0003−3907−3357]]

Faculty of Computer Science and Management, Wroclaw University of Science and Technology, Poland
michal.kedziora@pwr.edu.pl

Abstract. The purpose of this paper is to analyze the security and scalability problems occurring in private permissionless blockchain systems. The consent management system (CMS) based upon Hyperledger Fabric (HLF), was implemented in the selected blockchain-as-a-service (BaaS), and therefore led to consent-as-a-service (CaaS) deployment. The experiments results assessed to what level the network transaction throughput is affected by changing the world state size, which indicates scalability of chosen blockchain system implementation. Additional experiments with the IBM Blockchain Platform and the FastFabric framework (a HLF modification) were performed to prove the possibility to achieve transaction throughput comparable to the Ethereum blockchain network.

Keywords: Blockchain scalability · Permissionless blockchain · Blockchain security

1 Introduction

Utilising blockchain undergoes in several business areas. For example, in supply chain management (SCM) it enables to track product or service subsequent states in the business flow. Business partners can join the blockchain network to either read or write information to the ledger history. Blockchain properties (security, immutability) were found suitable for creating CMS [32,11,31,2,16,8]. However, the main and inherent limitation to the technology to overcome is scalability. A relatively new idea among cloud vendors is embracing blockchain to an offer. Such solutions are named Blockchain as a Service (BaaS). In this paper, BaaS term refers to every kind of service directly supported by the cloud services provider, including those available in their marketplace. BaaS is already used by well-known companies, mostly for the SCM purpose and seems to be applicable in a narrow target group for specifically defined requirements. However, there is a gap in research over BaaS since the idea of providing the blockchain network infrastructure, as well as the technology itself, is relatively new. None of them discussed related works considers the blockchain services in detail. The offers from network cloud providers tend to be treated collectively without testing solutions and without distinguishing their level of support.

Our paper addresses this gap by first surveying security and scalability problems occurring in private permissionless blockchain systems, as well as solutions to them, and then recreating a scalable blockchain system using several cloud environments. The emphasis is put on the blockchain systems hosted by cloud vendors in the form of BaaS. The currently available solutions offered by the most appreciated cloud providers are reviewed. The most promising services are tested for the real deployment of the CMS described by Agarwal et al. [2]. Implementing the CMS atop BaaS leads to CaaS. Through experiments, the proposed system's replication ability and its scalability are examined, along with assessing the feasibility of the CMS development in the provided cloud environment. An additional objective is to provide theoretical knowledge by introducing blockchain technology and performing a survey on its scalability, security and the emerging blockchain cloud services.

Paper structure is as follows. Blockchain technology is introduced in Section 1. Section 2 is a comprehensive survey which gathers information on various blockchain types, their favourable properties and main concerns. It shows and assesses the problems and solutions to scalability and security of blockchain systems and the current state of the distributed ledger technology (DLT) software. The possible and real-world blockchain use cases are analysed with paying particular attention to Consentio CMS whose originators are Agarwal et al. [2]. Section 3 presents results obtained by performing the scalability experiments on the CMS in the selected BaaS platform and its findings. The experience gained through testing various cloud services is summarised by estimating its costs, identifying the encountered problems and key differences. In Conclusions, the main assumptions of this research are pointed out and recapitulated, with identifying some future directions.

2 Related Work

With the growth of interest in blockchain technology from companies and institutions, the need for privacy and the ability to decide who can participate in a blockchain-based network arisen. Since Bitcoin and other cryptocurrencies used a completely public network, private and regulated solutions have been proposed. Hence the new terms introduced were permissioned as opposed to permissionless, as well as a private and public dichotomy. Not all of the research papers make a distinction between the four types of blockchain. Some, like [30,26], consider only private and public or permissioned and permissionless as two leading blockchain categories. Another prevalent approach is associating public blockchain with permissionless, and private with permissioned [21,17,24,20]. However, some recent publications [5,14,1] do distinguish those and acknowledge the existence of public permissioned and private permissionless blockchains as well, or just use a general *hybrid* term to categorise those recently emerging blockchains [21].

The Ethereum team (with "Vitalik" Buterin) introduced a *Scalability Trilemma* which states that an ideal blockchain system should have three characteristics: scalability, security and decentralisation. According to the trilemma, a blockchain system can have only two out of the three, e.g., improving scalability reduces a security level or the decentralised network on-chain [15]. There is a trade-off between getting a higher degree of one property and sacrificing the other ones. The concept has probably derived from the CAP theorem which applies to distributed systems, thus for blockchain as well. CAP stands for Consistency, Availability and Partition Tolerance. Any distributed system can have at most two of these aspects. It indicates that there cannot be a system that simultaneously: provides the same view of data to all the nodes, always responds to a user's request and works as expected despite the arbitrary physical network partitioning triggered by network failures.

Xie et al. [29] address the blockchain scalability issue to three fields: throughput, storage and networking. Throughput in the public blockchain is many times lower compared to the traditional payment methods. In 2013, Visa had the capability of handling more than 24000 transactions per second (TPS) [27]. Four years later it had the capacity of over 65000 TPS [28]. Meanwhile, Bitcoin can only process 3.3-7 [7] and Ethereum 7-15 transactions in the same amount of time [15]. It is worth noticing that the numbers are rather theoretical since they determine network capabilities – not the average TPS. As of December 15, 2019, Ethereum Blockchain Explorer shows an average of 7.2 TPS [12] and Ethereum's co-founder, Buterin, states that it will not scale to more than 15 TPS.

A limited throughput for Bitcoin is caused by its block size bound to 1 MB [4] and a long block interval time, i.e., time of confirming a transaction, including it in a block. An expected latency is 10 minutes [7] to create sufficient security. For comparison, another leading credit card company, Mastercard, claims to have the transaction processing speed below 500 ms [19]. Public permissionless blockchains like Bitcoin or Ethereum tend to use proof-of-work as a consensus mechanism. The resulting latency is the cost of propagating block over the decentralised network and utilising a relatively expensive consensus protocol, which is required to prevent Sybil attacks (e.g., 51% attack). An increasing block size seems to be an obvious solution to this problem. However, it results in a higher computing power requirement for confirming transactions, which leads to the risk of network centralisation by supercomputers.

Another point made by Xie et al. is storage scalability. A conventional blockchain system requires a node to store the complete transaction history. A current Bitcoin's blockchain size reaches 240 GB [6]. Therefore, a full node requires vast amounts of storage. The last scalability issue, related to networking, is the data transmission delay. In the Bitcoin model, each transaction broadcasts twice – first after creation (and moving to a transaction pool) and second after transmitting within a mined block.

Often omitted is the fact that the scalability issue should be addressed primarily to permissionless blockchains [29]. A transaction approval usually takes a few minutes. The problem is worse in public permissionless since every node is a validator. In public permissioned, as in any private blockchains, only selected nodes can validate. Permissioned blockchain systems can perform significantly better, sometimes even close to permissioned networks, such as Visa, Mastercard or PayPal, due to a low number of nodes and utilising different consensus algorithms. Latest Corda Enterprise release achieved 2580 TPS [22], while Hyperledger Fabric was once tested by IBM researchers to perform 3500 TPS [25,3] and others did manage to increase the throughput up to 20000 TPS [13] by proposing some architectural changes to the platform to reduce bottlenecks. The numbers are promising, but permissioned blockchain relies on having a trusted authority. Increasing both scalability and security violates decentralisation by allowing a middle-man, which seems to be just what Scalability Trilemma addresses. Permissioned blockchains are controversial because the original blockchain assumptions put an emphasis on the idea of having a purely distributed system, and their existence seems to be a step back to centralisation. Holochain, with its *agent-centric* approach, even claims not to be blockchain, but a decentralised technology (a P2P application framework) utilising hash chains [9]. Major scalability solutions try to improve research areas such as network efficiency, storage, data usage and a consensus algorithm [10].

Public blockchains suffer from many scalability issues, and the private ones tend to be centralised by an intermediary. Permissionless is not secure, since it allows nodes to be anonymous and permissioned may lead to centralisation as well. Another challenge in the blockchain system is security and data privacy. While decentralisation is a desirable feature, it leads to trouble in maintaining data integrity. Timeliness of transactions can affect both system's performance and security. One of the possible attacks is double-spending – a data integrity violation that may appear in purely distributed peer-to-peer (P2P) systems like blockchain. Malicious peers are a threat to the network, and even there are machine learning (ML) efforts made to detect them [23]. Allowing peers to verify new transactions is a way to prevent transferring ownership more than once. However, to verify transactions, their log has to be transparent. One of the most prominent blockchain technical limitations addressed by [10] Drescher is lack of privacy. The system has to reconcile two opposite concepts: transparency and privacy. Hence, it is not an appropriate solution to use cases that necessitate a high level of privacy. If a solution is needed for homogeneous environment where everyone trusts each other and has a full control over a flow of data, and the environment itself is not exposed to any external threats – blockchain is going to be a very slow and disturbing database. The technology becomes relevant if dealing with a lack of trust in the network. Then the responsibility is spread over many members, and each of them has a reason not to let the system break (either by an attack or by entering incorrect information). Moreover, even when discrediting some servers, there is always a dozen or more servers that allow regaining balance. However, convincing competitors to rely on technology and entrust an intellectual property to machines is challenging. That is why consortia are created. It is the authority and independence of the leaders that guarantee comfortable conditions for solving problems – regardless of market relations.

Drescher perceives the blockchain security model to be yet another limitation. Users identification, authentication and authorising their transactions require a pair of keys. A public key is an account number or, in terms of cryptocurrency, a wallet address. A private key is used to generate a unique digital signature to confirm *who* makes the transaction. Blockchain uses powerful cryptography methods. Its security model is not a problem itself. Unlike centralised applications in the P2P system, there are no security procedures to revoke access to the account when one looses their own private key. The situation is not rare. Most users store their private keys on the computer where it is prone to be stolen either by malware or hard disk failure [18]. Few ways to avoid losing the key is creating its backup or using offline storage. If the user fails to keep the key in a safe place, there is no way to reset it like most real-world systems allow to do with a password.

3 Experiments

Besides blockchain's widespread use in cryptocurrency, its attributes for private permissionless variation seem especially suitable for CMS. Agarwal et al. [2] presented in their article a CMS named *Consentio* implemented in Hyperledger Fabric. The need to track and manage consent to private data is considered in three areas: gathering electronic health records, smart infrastructure (smart cities) and within social media applications. The authors have emphasised creating a scalable system deploying blockchain back-end for CMS which was not within the scope of prior studies in this area. For this reason, it was decided to reproduce Consentio blockchain network and experiments. The proposed system is promising, because of:

- the ability to translate complex requirements of CMS to Fabric key-value world state;
- achieving high transaction throughput and making the system easy to scale for deployments (where an increasing number of individuals and resources defines scalability);
- ensuring low latency and therefore preventing double-spending by disallowing to have two or more transactions with the same key in the same block.

Despite the listed favourable properties, Agarwal et al. [2] made the article easy to reproduce thanks to precise descriptions and by providing the smart contract source code in their study. Transaction throughput was evaluated locally through micro-benchmarking on five servers connected by a switch. With all the details given, it should be possible to move the Consentio infrastructure involving a single endorser, orderer, peer, four clients and smart contracts for Individual-oriented World State (IWS) world state design into a cloud environment.

3.1 Testing Environment

Consentio comes along with sample implementation in Hyperledger Fabric. Agarwal et al. [2] has emphasised creating a scalable system deploying blockchain back-end for CMS, what was not a scope of prior studies in this area.

The aim is to implement CMS atop BaaS and therefore creating CaaS using Consentio chaincode. The main requirement for Consentio is Hyperledger Fabric framework, which narrows the selection of possible blockchain cloud services. Three cloud platforms that fulfilled the requirements will be compared: AKS (Azure), AMB (AWS) and IBM Blockchain Platform. With template solution from Microsoft Azure, the deployment failed at the beginning due to unknown error (no information was provided in the correlation ID). With Amazon's fully-managed service it was possible to build a simple network. However, in experiment planning, the new requirement for the system was discovered. One of the operations from Consentio chaincode needs a CouchDB backend database for a peer. As opposed to LevelDB that operates faster, this state database permits rich queries of data if the data has been modelled in a smart contract as JSON. AMB does not support CouchDB, even though HLF incorporates it. Ultimately, IBM Blockchain Platform could handle both deployment and experiments.

In Figure 1 the results of continuous writing 100 transactions by four local users are shown. They can be regarded as a measure of infrastructure performance. Each transaction held just one key-value pair. The measured time is a period between the earliest and latest created transaction in a single block created. 20000 with one value per each key have been sent. The world state key space (a number of key-value pairs in the registry) reached size of 11280.

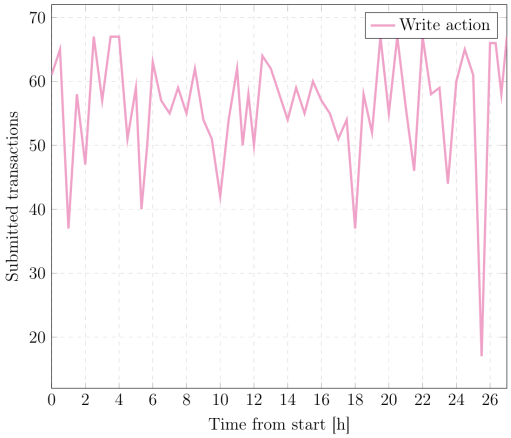

Fig. 1. Percent of submitted transactions in a single block (block size 100)

A single time measurement for TPS calculation was done differently, which considers time for creating a single block. It includes a range from creation of the first block to a last timestamp of submitted transaction. Figure 2 present accordingly read and write actions. With differences marked in Table 1, they correspond to Figures 6 and 7 from Consentio.

Table 1. Differences in volume between Consentio system and reproduction experiments

	Original	Reproduction
World state key space	20000	4000
Keys added in transaction	100	1
Value space per key	1; 100; 500; 1K; 5K; 10K	1; 2; 3; 4; 5; 6; 7; 8; 9; 10
Key space per transaction	100	10
Block size	100	20
Sent transactions	100000	4000

Red dashed line determines the average throughput which for *read* reached 1.02 and for *write* 1.55 TPS. With the statistical significance of 5%, margins of error equal:

- Read: 1.0244 ± 0.0297 TPS.
- Write: 1.5395 ± 0.1110 TPS.

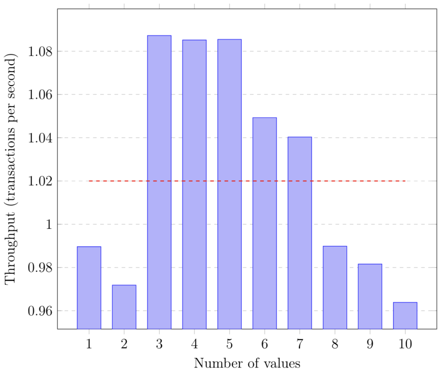

Fig. 2. Read throughput performance vs. size of value space (keys touched per transaction is kept constant at 100 and key space is kept constant at 4000)

3.2 Results

With proper setup to replicate a write throughput experiment, the testing script for submitting transactions using four users, each one having 25 threads (100 threads total), has been run. The IBM Blockchain Platform infrastructure could not handle such overload. Node.js Fabric SDK returned errors caused by the inability to contact the peer. Web UI available through the console was nearly unreachable at the time of sending transactions. Approximately 56% of written transactions were appended to each block, when others were lost due to insufficient resources. Ultimately, the world state key space reached a size of 11280 instead of 20000.

Some of Consentio experiments were repeated though (on the new channel, but for lower volumes, which are pointed out in Table 1). Among the three designs, the IWS world state was tested. Besides, the infrastructure still differs from the original one. Except for using IBM Kubernetes free cluster with 4 CPU and 2 GB of RAM to host the blockchain network, the four clients have been run on the MacBook Pro 2019 with Intel Core i5-8279U 2.4 GHz (4 cores) and 16 GB of RAM. The Internet connection used to communicate with the blockchain network was symmetrical with the download rate of 8.1 Mbps and 8.0 Mbps upload. The client application utilises Fabric SDK in Node.js to connect to HLF v1.4 blockchain network set up on IBM Blockchain Platform. Agarwal et al. [2] did not mention the client application SDK used for their experiments.

Considering experiments the replication and the first attempt of Consentio CMS reproduction for the same volume for which achieved transaction throughput was lower than in Bitcoin network (3.13 TPS) with 56.4% rate of successfully submitted transactions, insufficiencies in three areas have been examined compared to the Consentio network: physical resources, a blockchain framework and a network connection.

Table 2. Number of unsuccessful transactions during read or write to the world state containing a certain number of values per key

Values	1	2	3	4	5	6	7	8	9	10	Average
Read	1	5	0	0	1	0	1	3	0	0	1.10
Write	2	2	0	0	1	0	0	7	2	3	1.55

In the case of physical resources, the free tier offered by IBM cloud vendor could not handle the desired traffic. Another considerable difference gap concerns the difference in blockchain framework used – in the study it was plain HLF. In their experiments, Agarwal et al. [2] used FastFabric, which was mentioned earlier. Gorenflo et al. [13] introduced this HLF v1.2 modification in 2019. At the time, it enabled improving the transaction throughput from 3000 TPS (for baseline HLF v1.2) to 20000 TPS, which was shown through benchmarking. Assuming the improvement resulting from FastFabric to scale linearly, Consentio deployed in HLF would achieve 6-7 times fewer TPS – presumably around 900 TPS on average for both reading and writing. The gap between the original and reproduction experiments results is not significantly diminished when applying the same calculations for the obtained throughput. TPS of 7.7 for reading and

10.85 for writing are the highest values likely with FastFabric. The last limitation is network connection.

In Consentio experiments, 1 Gbit/s switch connected the servers. The isolated environment guaranteed comparatively a stable level of exchanged data. In reproduction, the requests were sent through an Internet connection with approximately 125 times worse throughput. Also, every request was sent from Wrocław (Poland) to the cluster in the city of Dallas (United States). The connection type and the distance led to high latency. As claimed in the Consentio paper [2], the proposed CMS is replicable and indeed scalable. Physical testing infrastructure was almost adequately efficient to handle requests of the selected volumes. Minor reservations relate to occasional errors, which resulted in failing to add a transaction or in an inability to read the world state. *Lost* transactions for both read and write corresponding to the value iterations in Figure 2 are presented in Table 2. Lost transactions account for roughly 0.03% of the total transactions made.

4 Discussion

Initially, for Consentio, all three experiments aimed to be repeated. In reproduction, key differences are physical infrastructure (that is much less technically advanced) and using plain HLF v1.4 instead of FastFabric [13] framework. Also, transaction times have been expected to increase because of the ongoing endorsement process. Despite the differences in physical infrastructure, the framework used, the presence of Endorser, these factors should not be meaningful in reproduction. Consentio experiments prove that the proposed system is scalable. With obviously decreased TPS values, the same measurements have been expected to show the linear trend.

Despite significant fluctuations of the obtained TPS values occurring in the results of actual reproduction, they do not seem to correlate to the increasing value space of the world state. However, TPS might have been affected by the network connection. The throughput reaches its overhead in the middle of Figure 2. The number of values ranging from 3 to 7 gave results above the average in both cases. Measurements for the interval were done at night between 11 PM and 9 AM the following day. Reduced internet traffic at the time could have had an impact on them.

Regarding the cloud blockchain solutions, the definition of BaaS is yet vague. There are different levels and types of support for DLT. For instance, a blog post describing a simple blockchain network deployment (that might even come as a configured Docker container) using any web infrastructure needs to be distinguished from fully-managed blockchain services. These usually comprise extensive documentation, web UI, online video or interactive tutorials, and technical support.

When deciding on certain BaaS, one usually has to seek precise information about frameworks included. Like in the case of Amazon QLDB, their versions are rarely mentioned explicitly until creating a final configuration to deploy or paying attention to hyperlinks to framework's documentation in the deployment guide.

BaaS customers of the discussed platforms prefer to use permissioned blockchains for SCM. Due to blockchain's technology data-centric approach, it is especially convenient for multistakeholder governance.

The key results of this empirical study are:

- Recreating the Consentio blockchain network was fully possible with one of the tested BaaS platforms, the IBM Blockchain Platform.
- Using the cloud environment and a similar configuration to Consentio with four client users (one hundred threads total) continuously writing transactions to the blockchain, it took about 27 hours to get a key space of 11280. During that process approximately only 56% of sent transactions were submitted due to insufficient resources.
- For overall lower volumes, the experiment for reading consent had average of 1.02 TPS and 1.55 TPS for writing. However, with FastFabric framework (a HLF variation), the highest values likely are estimated to be 6-7 higher, i.e. 7.7 TPS for reading and 10.85 TPS for writing.
- The results suggest that the network transaction throughput is not affected by changing the world state size, which proves the Consentio CMS to be a scalable system in this sense.

5 Conclusion and future work

Our paper aimed to fill the gap in the research over BaaS by making the following contributions. First by implementing a scalable blockchain system in a cloud environment. Second by surveying security and scalability problems occurring in private permissionless blockchain systems and solutions to them. The most promising services have been tested for the real deployment of the CMS, and led to creating CaaS. Through experiments, the proposed system's replication ability and its scalability have been examined, along with assessing the feasibility of the CMS development in the provided cloud environment. Considering the experiments replication, insufficiencies in three areas have been examined: physical resources, a blockchain framework and a network connection. All the factors combined led to high latency in transaction throughput for both reading and writing. Using the cloud environment and a similar configuration to Consentio with four client users (one hundred threads total) continuously writing transactions to the blockchain, it took about 27 hours to get a key space of 11280 with approximately 44% of lost transactions due to limited resources. For overall lower volumes, the experiment for reading consent reached an average of 1.02 TPS and 1.55 TPS for writing. However, with FastFabric framework (a Hyperledger Fabric variation), the highest values likely are estimated to be 7.7 TPS for reading and 10.85 TPS for writing which is a similar throughput to Ethereum cryptocurrency. Nevertheless, the throughput results obtained are thousands of times lower, which indicates the insufficiency of the provided infrastructure. Future directions indicate that more advanced technical infrastructure for Consentio deployment would enable to reproduce all the experiments for the same volumes. In IBM Blockchain Platform, investing in a more powerful Kubernetes cluster should improve the throughput results. Independently, adding more peers to the blockchain network would increase the ability to endorse the transactions (assuming endorsement policy that does not require all peers).

References

1. Acharya, V., Yerrapati, A.E., Prakash, N.: Oracle Blockchain Quick Start Guide: A practical approach to implementing blockchain in your enterprise. Packt Publishing Ltd (2019)
2. Agarwal, R.R., Kumar, D., Golab, L., Keshav, S.: Consentio: Managing consent to data access using permissioned blockchains. arXiv preprint arXiv:1910.07110 (2019)
3. Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y., et al.: Hyperledger fabric: a distributed operating system for permissioned blockchains. In: Proceedings of the Thirteenth EuroSys Conference. p. 30. ACM (2018)
4. Badr, B., Horrocks, R., Wu, X.B.: Blockchain By Example: A developer's guide to creating decentralized applications using Bitcoin, Ethereum, and Hyperledger. Packt Publishing Ltd (2018)
5. Belotti, M., Božić, N., Pujolle, G., Secci, S.: A Vademecum on Blockchain Technologies: When, Which, and How. IEEE Communications Surveys & Tutorials **21**(4), 3796–3838 (2019)
6. Blockchain Luxembourg: Blockchain Size (2017), https://www.blockchain.com/charts/blocks-size. Last accessed 14 Sep 2019
7. Croman, K., Decker, C., Eyal, I., Gencer, A.E., Juels, A., Kosba, A., Miller, A., Saxena, P., Shi, E., Sirer, E.G., et al.: On scaling decentralized blockchains. In: International Conference on Financial Cryptography and Data Security. pp. 106–125. Springer (2016)
8. Dias, J.P., Ferreira, H.S., Martins, Â.: A blockchain-based scheme for access control in e-health scenarios. In: International Conference on Soft Computing and Pattern Recognition. pp. 238–247. Springer (2018)
9. Donath, M.: Holochain Docs (2019), https://developer.holochain.org/docs/what-is-holochain/what-is-holochain. Last accessed 06 Jan 2020
10. Drescher, D.: Blockchain Basics: A Non-Technical Introduction in 25 Steps. APRESS, New York (2017)
11. Ekblaw, A., Azaria, A., Halamka, J.D., Lippman, A.: A Case Study for Blockchain in Healthcare:"MedRec" prototype for electronic health records and medical research data. In: Proceedings of IEEE open & big data conference. vol. 13, p. 13 (2016)
12. Etherscan: Ethereum (ETH) Blockchain Explorer (2020), https://etherscan.io. Last accessed 25 Apr 2020
13. Gorenflo, C., Lee, S., Golab, L., Keshav, S.: Fastfabric: Scaling hyperledger fabric to 20,000 transactions per second. In: 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC). pp. 455–463. IEEE (2019)
14. Huertas, J., Liu, H., Robinson, S.: Eximchain: Supply Chain Finance solutions on a secured public, permissioned blockchain hybrid. Eximchain White Paper **13** (2018)

15. Hummer: Sharding FAQ (2017), https://github.com/ethereum/wiki/wiki/Sharding-FAQ. Last accessed 12 Dec 2019
16. Jesus, V.: Towards an accountable web of personal information: the web-of-receipts. IEEE Access **8**, 25383–25394 (2020)
17. Lai, R., Chuen, D.L.K.: Blockchain–from public to private. In: Handbook of Blockchain, Digital Finance, and Inclusion, Volume 2, pp. 145–177. Elsevier (2018)
18. Malanov, A.: Problems and risks of cryptocurrencies (Nov 2017), https://www.kaspersky.com/blog/cryptocurrencies-intended-risks/20034/. Last accessed 24 Feb 2020
19. Mastercard International Incorporated: Mastercard Transit Solutions Guide (2018), https://graphic.mastercard.com/acquirer-newsletter/issue12/pdf/14-f-mastercard-transit-solutions-guide.pdf. Last accessed 14 Dec 2019
20. Novotny, P., Zhang, Q., Hull, R., Baset, S., Laredo, J., Vaculin, R., Ford, D.L., Dillenberger, D.N.: Permissioned blockchain technologies for academic publishing. Information Services & Use **38**(3), 159–171 (2018)
21. Onik, M.M.H., Miraz, M.H.: Performance Analytical Comparison of Blockchain-as-a-Service (BaaS) Platforms. In: International Conference for Emerging Technologies in Computing. pp. 3–18. Springer (2019)
22. R3 Limited: Sizing and performance – Corda Enterprise 4.3 (2018), https://docs.corda.r3.com/sizing-and-performance.html. Last accessed 22 Dec 2019
23. Rahouti, M., Xiong, K., Ghani, N.: Bitcoin concepts, threats, and machine-learning security solutions. IEEE Access **6**, 67189–67205 (2018)
24. Sánchez, D.C.: Raziel: Private and verifiable smart contracts on blockchains. arXiv preprint arXiv:1807.09484 (2018)
25. Schatsky, D., Arora, A., Dongre, A.: Blockchain and the five vectors of progress. Deloitte Insights (2018)
26. Swanson, T.: Consensus-as-a-service: a brief report on the emergence of permissioned, distributed ledger systems. Report, available online (2015)
27. Visa Inc.: The technology behind Visa (2013), last accessed 14 Dec 2019
28. Visa Inc.: Visa Fact Sheet (2017), last accessed 14 Dec 2019
29. Xie, J., Yu, F.R., Huang, T., Xie, R., Liu, J., Liu, Y.: A Survey on the Scalability of Blockchain Systems. IEEE Network **33**(5), 166–173 (Sep 2019). https://doi.org/10.1109/MNET.001.1800290
30. Yaga, D., Mell, P., Roby, N., Scarfone, K.: Blockchain technology overview. arXiv preprint arXiv:1906.11078 (2019)
31. Zhang, Y., Kasahara, S., Shen, Y., Jiang, X., Wan, J.: Smart contract-based access control for the internet of things. IEEE Internet of Things Journal **6**(2), 1594–1605 (2018)
32. Zyskind, G., Nathan, O., et al.: Decentralizing privacy: Using blockchain to protect personal data. In: 2015 IEEE Security and Privacy Workshops. pp. 180–184. IEEE (2015)