Computing and Informatics, Vol. 40, 2021, 880–896, doi: 10.31577/cai_2021_4_880

CLOUD SOLUTIONS FOR PRIVATE PERMISSIONLESS BLOCKCHAIN DEPLOYMENT

Hanna Grodzicka, Michal Kedziora, Lech Madeyski

Department of Applied Informatics
Wroclaw University of Science and Technology, Poland
e-mail: {michal.kedziora, lech.madeyski}@pwr.edu.pl

Abstract. This paper aims to survey the security and scalability problems occurring in private permissionless blockchain systems and solutions to them. The emphasis is put on the blockchain systems hosted by cloud vendors in the form of Blockchain-as-a-Service (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 consent management system (CMS). Implementing the CMS atop BaaS leads to creating Consent-as-a-Service (CaaS). Through experiments, the proposed system's replication ability and its scalability are examined, along with assessing the feasibility of the consent management system development in the provided cloud environment.

Keywords: Blockchain, blockchain-as-a-service, permissionless blockchain

1 INTRODUCTION

Nowadays, web cloud services are gaining popularity. The goal is to deliver the whole cloud infrastructure to the customer. According to the Gartner Research made by [38], the leading companies in the market are AWS, Microsoft and Google. Alibaba Cloud, Oracle and IBM are yet said to be niche vendors in this area. One of the recently emerging services provided in the cloud computing model is blockchain technology. The public provision of such services is referred to as BaaS in the literature [31, 24]. Depending on whether hosting a blockchain network or its peer client, it is perceived respectively as Platform-as-a-Service (PaaS) or Software-as-a-Service (SaaS) variant.

BaaS incorporates delivering physical resources, blockchain implementation, and set-up templates, allowing individuals and enterprises to adopt blockchain technology fast by reducing the overall process complexity. It is also important as blockchain technology is issued with many security concerns [23, 37, 22]. The main goal is to make application deployment easier by delegating a set of responsibilities and necessary technical knowledge connected to blockchain and cloud infrastructure to the third-party provider. The underlying platform should not be a concern when using BaaS. Therefore, a user can focus on adding a business value to the developed software. Implementing an own application using BaaS is highly dependent both on the cloud vendor and the blockchain framework it utilises. A framework is a code that defines the principles by which blockchain is created – a programming language, approach to smart contracts, a consensus mechanism, and what requirements should the participants meet. Some frameworks are more suitable for areas such as the supply chain, Internet of Things (e.g., Hyperledger Sawtooth and Hyperledger Burrow), mobile applications (Hyperledger Iroha) or identity management (Hyperledger Indy).

We assess and test recently emerging BaaS solutions hosted by the most appreciated cloud vendors. For this purpose, we develop a research cloud-based blockchain infrastructure with Hyperledger Fabric (HLF) and reproduce Consentio scalable CMS proposed by Agarwal et al. An attempt to differentiate types of BaaS is made, which contributes to filling a gap in the research. Deployment attempts are made for Amazon Managed Blockchain (AMB), Azure Kubernetes Service (AKS) and IBM Blockchain Platform. The experiments focus on measuring transaction throughput and solutions' scalability. The results can be utilised in the deployment of the proposed CMS or any blockchain network of choice in a selected cloud environment. The whole research is meant to be reproducible by explaining the deployment flow and providing precise instructions.

Paper structure is as follows. Distributed Ledger Technology and BaaS are introduced in Introduction. Next section (Related Works) analyses blockchain use cases, and the current state of knowledge in the paper research area. Section 3 introduces the methodology of research. The current state of the DLT and BaaS is examined in Section 4 and also summarises, interprets, and discusses the results of testing various cloud services. The conclusion points out and recapitulates the main assumptions of this study.

2 RELATED WORKS

Although blockchain's main and still leading use case is cryptocurrencies, it gained popularity in other areas. Dymek et al. [8] conducted research that points out that besides traditional banking and trading usage, blockchain applications are rarely known even among computer science students. Some research papers try to use blockchain technology in favour of Internet of Things (IoT) [36, 39, 14]. Blockchain enables many parties to contribute to one tamper-proof data source. Information can be centralised and exchanged effortlessly between multiple devices. Combined with cryptography employed, it makes the whole environment highly resistant to fraud, when a device is hacked.

Blockchain's type determines the area which it can be applied to. Use cases for all the four blockchain types are presented in Table 1. When creating a medical data storage system for a hospital, it is clear that only doctors should be able to add notes and results to the system. Similarly, data reading would be limited only to the patient and his doctors. Private permissioned blockchain could be employed in the situation. The choice of a suitable blockchain platform depends on which of the three characteristics, i.e., scalability, security and decentralisation mentioned in the Scalability Trilemma [16], are the most favourable. Business-to-business (B2B) markets find decentralisation the least important feature. While public permissionless resigns itself to scalability.

For enterprises, the use of private blockchain became widespread among other types. Taking into account all blockchain benefits and how quickly cryptocurrency has gained popularity, ideas such as an instant payment network or a lightweight financial system (with relatively low stakes) might seem suitable. However, the main problem is confidentiality. The transaction should not be visible to other network participants, especially when making payments or exchanging confidential assets. Even without the exact data being visible, it is possible to correlate information.

Blockchain technical counterpart to handle the same type of problems is a (traditional) centralised database, which is the other most apparent use case. In addition to providing a single view of the entire system, which is not owned by any particular party, it characterises robustness on the level of multiple organisations. There is no need to have a backup system. Lost data can recover from other network participants. Yet again, with a distributed P2P shared database, the difficulty is to ensure confidentiality. All the data is broadcast in the network. In the case of a centralised database, the same information would be visible only to an intermediary and involved parties. Estimating the performance of the two solutions might not seem obvious, but in terms of processing transactions, a centralised database is faster because it has to do less. In a blockchain, a node has to do the same processing and additionally verify cryptographic signatures, and might need to spend time to establish consensus, etc. Hence, choosing between a blockchain and a centralised database is a matter of deciding between acceptable trade-offs.

However, there are situations where a golden mean is needed. Either everyone has access to data, but only a limited group can add information, or vice versa – anyone can add it, but only the privileged has the right to read the data. Examples are the passport system or the diploma system at schools. Public permissioned blockchain can handle these situations. Such solutions have one more advantage. Because participants are qualified – although they can be anonymous but need to meet definite rules – the mechanism that is used to reach a consensus may be less rigorous. This means Bitcoin Proof-of-Work withdrawal (which uses vast amounts of energy to add each block) and opening to new protocols, e.g., Proof-of-Elapsed-Time (PoET) [5] or Yet Another Consensus (YAC) [30].

	Permissionless	Permissioned
Public	• cryptocurrency • video games	• voting • poll records • land titles • university degrees
Private	• supply chain provenance • government record keeping assessor records	• instant payment network • database • multijurisdictional process • lightweight financial system • notarising and timestamping • tax returns • consortium and federations • medical records

Table 1. Use-cases for specific types of blockchain (source: [13])

Using blockchain became particularly popular in supply chain management (SCM) [35]. The idea is to track a physical asset as it moves across many organisations to a client, and therefore to monitor the whole workflow and environmental conditions. With blockchain, the process results in improved security and gives real-time insights. That is how Starbucks can follow a coffee bean from a farm where it gets collected until reaching the barista who serves it. Each step of the travel (an event) can be submitted as a transaction. Similarly, blockchain serves as a digital traceability technology to GE Aviation, that follows aircraft parts, and Bühler tracking the journey of their crops from farms to markets. 3M, assuring that no one tries to counterfeit their labels by tracking products in the supply chain, has even come up with Label-as-a-Service (LaaS).

Among many other use cases such as financial platforms [15], online voting or even digital identity, Agarwal et al. [1] tries adapting blockchain for CMS. The idea is not new and was introduced before the [41, 9, 40, 21, 7]. However, Agarwal et al. focus is implementing a *scalable* system. 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.

3 METHODOLOGY AND TESTING ENVIRONMENT

The aim is to implement CMS atop BaaS and therefore create CaaS using Consentio chaincode. The main requirement for Consentio is Hyperledger Fabric framework, which narrows the selection of possible blockchain cloud services. Moreover, 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. Three cloud platforms are compared to the deployment.

	Reproduction	Original
World state key space	4 000	2 0000
Value space per key	1; 2; 3; 4; 5; 6; 7; 8; 9; 10	1; 100; 500; 1K; 5K; 10K
Key space per transaction	10	100
Sent transactions	4 000	100 000

Table 2. Differences in volume between Consentio system and reproduction experiments

All of Consentio experiments had a block size limit configured to 100 transactions. Thereat, the channel's advanced settings have been changed. All four block parameters shown were modified. For reproduction, the most important was *max message count* (i.e., the maximum number of transactions that can exist in a block before a new block is cut) set to 100. To ensure that the block size is determined by the number of transactions, other parameters have been set to their maximum values. The maximum size of any block (*absolute max bytes*) is 99 MB, and its preferred block size (*preferred max bytes*) is circa 79 MB. A decreasing *timeout* value can improve latency, but its excessive reduction may result in a lower throughput by not allowing the block to fill its maximum capacity. Since throughput is the tested factor (to be minimised), the timeout has been set to 5 minutes.

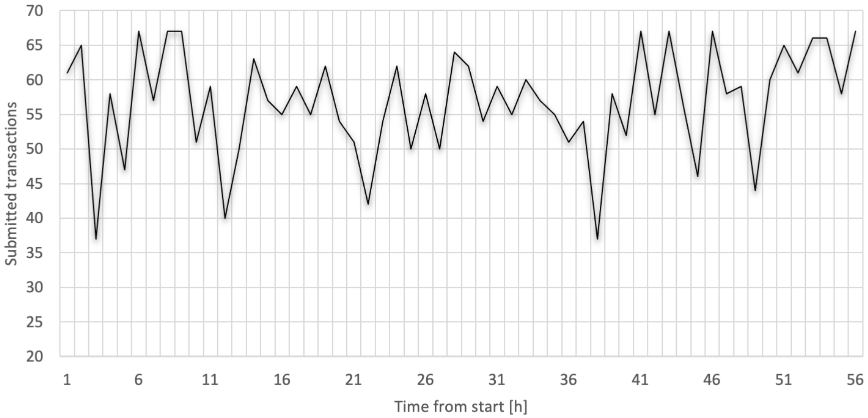


Figure 1. Percent of submitted transactions in a single block

The system presented in the article [1], named *Consentio*, comes along with a sample implementation in Hyperledger Fabric. Agarwal et al. have emphasised creating a scalable system deploying blockchain back-end for CMS, which was not the scope of prior studies in this area.

The results of writing transactions by four local users are shown in Figure 1 as a measure of infrastructure performance. The measured time is section between the earliest and latest transaction in a single block. During experiments, the world state key space reached a size of 11 280.

A single time measurement for transactions per second calculation considers the time for creating a single block and includes a range from creation of the first block to the last timestamp of the submitted transaction. Figure 2 and Figure 3 present accordingly read and write actions. The average throughput reached respectively 1.02 and 1.55 transactions per second.

4 ANALYSIS OF BLOCHAIN-AS-A-SERVICE

In this study, blockchain and ledger technologies available from web cloud providers are divided into three categories:

1. Fully-Managed Services.

   They are maintained by a service provider specifically for blockchain use cases. It requires the least technical knowledge to use them. Most configuration can be done via web UI.

2. Solution Templates from a Web Service Provider.

   Templates tend to use basic services such as VM with an operating system, delivered with pre-configuration and set-up documentation. It speeds up the deployment process but does nothing beyond the web service offers.

3. Solutions Validated by a Provider (Available at a Marketplace), but Introduced by a Partner.

   The concept is similar to templates, but liability lies on both sides. An independent software vendor (ISV) provides software in the SaaS model, a container or information required to launch and run the VM. The web service provider accepts and promotes the solution (in its marketplace) as well as delivers the infrastructure. Noteworthy, the solution available in the marketplace might belong to the provider itself.

The aforementioned are ordered descending by the level of required additional configuration and technical knowledge, but also by the possibilities. Advanced configuration is time-consuming, but gives more control, while fully-managed services may impose some specific architecture and limit the features available for the blockchain framework.

Except for services made especially for the blockchain purposes and these available in the marketplace, both AWS and Microsoft Azure support templates. Most of these solutions are based upon casual VM often incorporating Docker software. The overview of BaaS from the three leader web service providers compared to the blockchain frameworks they benefit from is in Table 3. It includes only fully-managed services (category no. 1) and solution templates (category no. 2).

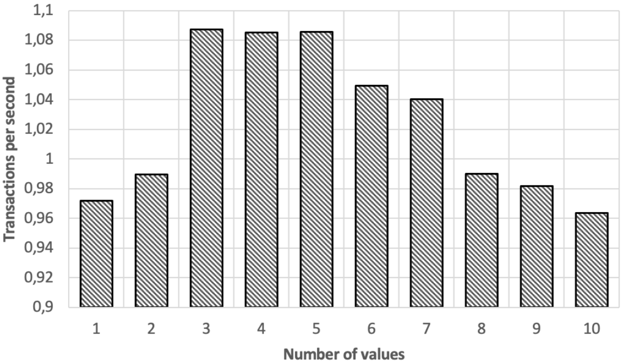


Figure 2. Read throughput performance vs. size of value space

AWS announced their latest BaaS on 2019-04-30 [4]. In AMB, if the blockchain network is built upon the Hyperledger Fabric framework, each peer node has to run *chaincode*, which is a Java or Go application enabling interaction with a ledger. Such applications are smart contracts and need to implement the Chaincode Interface [17].

A few large companies use AMB. AT&T Business telecommunication company found a use case for their IoT products. Automated data collection serves to track supply chain processes. Supply chains are also of main interest for Nestlé food corporation, to which AMB facilitates food and beverage tracking from the source to customer. The service enables facile collaboration with business partners on products history. Although supply chain management seems to be a dominant blockchain use case, an Asian investment holding company, Singapore Exchange Limited used AMB for the settlement procedure. With smart contracts, their DVP ensures that securities are delivered only after making the corresponding payment. Blockchain service before AMB is Amazon QLDB introduced in 2018. It claims to provide an immutable database for ledger-like applications. Behind the scenes, QLDB uses either Hyperledger Fabric or Ethereum. The two blockchain services can act together because it is possible to replicate transactions from AMB to QLDB. Except for those two fully managed services, blockchain network deployment templates are supported [2]. On Amazon ECS or Amazon EC2, one can set up Hyperledger Fabric or Ethereum. Furthermore, in AWS Marketplace, there are currently 74 products in the blockchain category. Among them, the vast majority (57 products) come as pre-configured Unix-based AMI and nine are said to be SaaS [3]. Among global partners, called APN, on Amazon's site has highlighted these solutions from category no. 3:

- Kaleido Enterprise Blockchain SaaS by ConsenSys [6].
- Corda by R3 in version 4.0, deployed on AMI [34].

Microsoft Azure offers a similar range of blockchain-related technologies to AWS. At Microsoft Azure portal, one service falls into the *blockchain* category, Azure Blockchain Service. In *others* category Blockchain Data Manager and Corda can be found though. Microsoft Azure Marketplace has 88 blockchain products [26]. Some of the most remarkable are:

- Azure Blockchain Workbench by Microsoft which comes with Ethereum PoAuth. It is compatible with Quorum and uses Solidity v0.5.10. Future integration involves Hyperledger Fabric and Corda [26].
- Quorum EEA v2.5.0 [28].
- Kaleido Enterprise Blockchain SaaS by ConsenSys [25].
- MultiChain on Azure by MultiChain [27].

Google Cloud does not have any blockchain category in its *solutions* nor *products.* In Google Cloud Platform Marketplace, 28 blockchain solutions can be found though, among which the most noteworthy are:

- Hyperledger Fabric and Composer by Google Click to Deploy. The image is deployed on GCE and runs Hyperledger Fabric v1.2 with Composer v0.20.1 [11].
- Ethereum by Google Click to Deploy. The image is deployed on GCE and runs Ethereum v1.8.12 [10].

IBM is a collaborator on the open-source Hyperledger umbrella project, entailing their fully-managed service to offer this framework. IBM Blockchain Platform is a SaaS that supports Hyperledger Fabric in version 1.4.6. IBM Cloud has put it into *the databases* category [19]. For smart contract and application development and deployment, IBM Blockchain provides Visual Studio Code extension [20]. There are other free resources for blockchain developers. Except for extensive documentation and a series of YouTube videos on getting started with the platform, there are two tutorials with an introduction to distributed ledgers and one article about the Hyperledger Fabric framework [18].

The service is free for 30 days preview if using IBM Cloud Kubernetes free cluster to deploy the platform. The Blockchain network is based upon a cluster created with IBM Cloud Kubernetes Service, which is a prerequisite since it should contain IBM Blockchain Platform components. The platform's documentation describes the solution architecture. With this enterprise solution, it is possible to deploy selected Fabric components such as peers, ordering service or Certificate Authority (CA). Nodes can be run on-premise or belong to any cloud environment.

Oracle in its Cloud Marketplace [33] has 14 applications (among which seven are free) and one service for Blockchain Platforms in the PaaS product category. None of these solutions is rated or reviewed. Oracle's PaaS that fulfils requirements for fully-managed service is Oracle Blockchain Platform Cloud Service. It is based on Hyperledger Fabric and since the platform's 19.2.3 release supports the 1.4.1

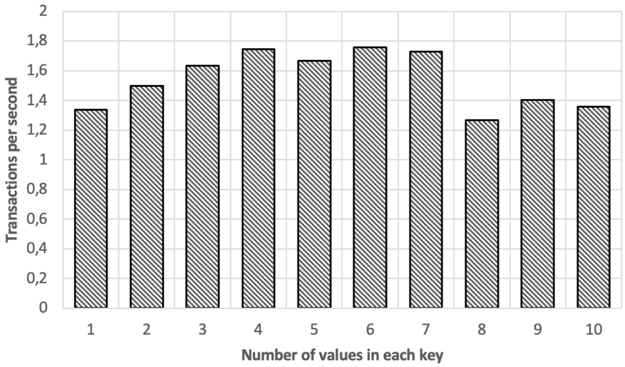


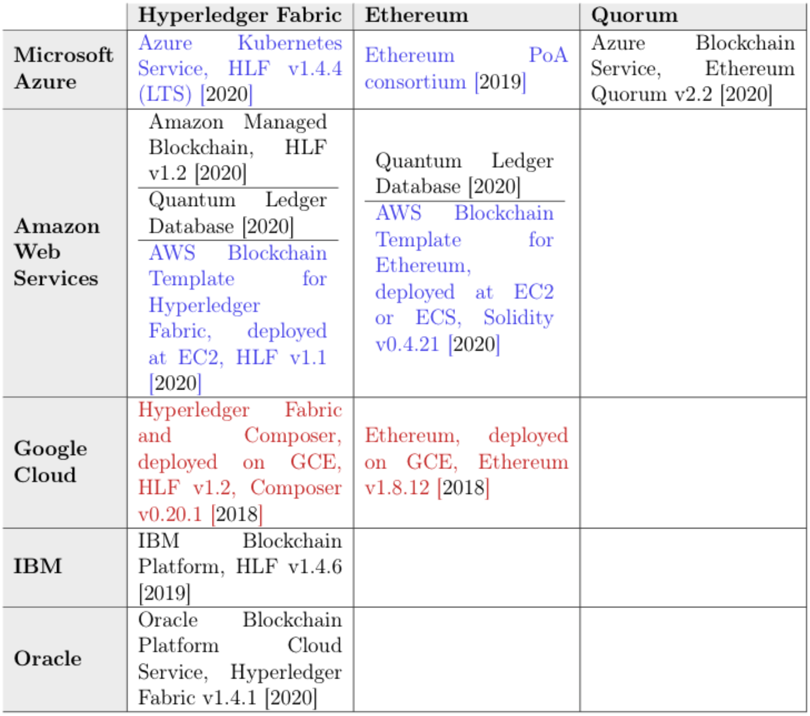
Figure 3. Write throughput performance vs. value size per key

version. The BaaS is said to be able to perform 500 queries per hour, which gives 8.3 transactions per second (TPS) throughput [32].

Templates (BaaS category no. 2) are marked *blue* and marketplace solutions (category no. 3) *red*. IaaS components that deploy the templates are mentioned along. It is worth noticing that AKS from Microsoft Azure and Blockchain Platform from IBM solutions are technically identical (Hyberledger Fabric deployment uses Kubernetes service underneath), but they differ in the level of support for the Hyperledger Fabric platform itself. That is why they fall into different service categories – AKS is a template while IBM blockchain platform is a fully-managed service. Marketplace solutions were not mentioned in the table except for these from Google Cloud, which are the only blockchain services offered by the cloud vendor itself.

5 RESULTS AND DISCUSSION

While using different blockchain services from the three cloud vendors, some problems, key differences and advantages were identified. The summary concerning services capabilities and the deployment process is made to evaluate them. Setting up AMB is a simple task for creating a network itself through legible and fault-tolerant UI. Despite relatively high prices, there are free tier options for selected instance types and a lucid expenses track and control system. When it comes to adding other network components and actions (e.g. enrolling identities, organisations, MSP, creating channels, chaincode installation), the vast majority of them are done through CLI. Lack of UI gives the impression that the service is yet underdeveloped. AMD offers only one framework, which is HLF with the release version 1.2, while the most recent one is 2.0. Referring to the peer database problem is described in the previous paragraph, it does not fully support all the features of the framework, even in such an old version.



	Hyperledger Fabric	Ethereum	Quorum
Microsoft Azure	Azure Kubernetes Service, HLF v1.4.4 (LTS) [2020]	Ethereum PoA consortium [2019]	Azure Blockchain Service, Ethereum Quorum v2.2 [2020]
Amazon Web Services	Amazon Managed Blockchain, HLF v1.2 [2020] Quantum Ledger Database [2020] AWS Blockchain Template for Hyperledger Fabric, deployed at EC2, HLF v1.1 [2020]	Quantum Ledger Database [2020] AWS Blockchain Template for Ethereum, deployed at EC2 or ECS, Solidity v0.4.21 [2020]	
Google Cloud	Hyperledger Fabric and Composer, deployed on GCE, HLF v1.2, Composer v0.20.1 [2018]	Ethereum, deployed on GCE, Ethereum v1.8.12 [2018]	
IBM	IBM Blockchain Platform, HLF v1.4.6 [2019]		
Oracle	Oracle Blockchain Platform Cloud Service, Hyperledger Fabric v1.4.1 [2020]		

Table 3. Fully-managed services and templates compiled with their providers and available blockchain frameworks with their versions (source: [13])

A full Consentio reproduction is not possible with the current AMD. Peer nodes would require to enable CouchDB instead of using the default built-in LevelDB state database to query a consent. The option is not available and is not announced to be supported any soon. With AKS, the deployment could not be completed despite multiple attempts. The information about the deployment failure fetched with the provided correlation ID was empty. The situation happened several times. On the contrary, using IBM Blockchain Platform enabled to create Consentio network.

Extensive documentation with descriptions, clear instructions, architecture and flow diagrams, and many other resources such as articles, text and video tutorials, code samples on GitHub repositories help with getting started with the platform. This BaaS solution is flexible because it enables to deploy only selected Fabric components such as peers, ordering service or CA. Nodes can be run on-premise or belong to any cloud environment. Adding and managing HLF components can be done through a simple UI. CLI usage is minimised. At each configuration step, the explanations for Fabric components can be found and sometimes the estimated time. When creating a new component, all required and optional fields have hints. Every component definition can be exported in a JSON file and uploaded as an existing definition instead of re-creating one, causing the configuration to be portable within IBM Blockchain Platform. The service could serve as an interactive educational resource because it is free, well-documented and provides information about each configuration.

During the research, we decided to analyse the expected minimal monthly costs for the BaaS (Table 4). The costs are estimated for only the chosen services from four cloud vendors (AWS, Microsoft Azure, IBM Cloud and Google Cloud). If prices were given only for a short period (e.g., an hour), they have been added up to get the monthly cost. One month is assumed to be 730 hours of the running environment. Furthermore, the most modest possible network is taken into calculation, i.e. with only one node and using VM for the most reasonable price. The currency used for the comparison is the United States Dollar (USD).

Service	**Estimated Cost (USD)**
Amazon Managed Blockchain	52.288
Azure Kubernetes Service	103.37
IBM Blockchain Platform	211.70
Google Cloud HLF	9.50
Google Cloud Ethereum	24.75

Table 4. Summary of estimated monthly costs for selected blockchain services

The exact calculations we based on the following assumptions: Amazon Web Services, as opposed to Google Cloud, do not have a present approximate cost, and customers have to calculate it themselves to estimate the price before deployment. AMB is considered in this section. Blockchain standard network edition is 0.30 $ per hour, CA storage rate is $ 0.10 GB per month, CA data written rate is $ 0.10 GB, peer node is $ 0.034 per hour and EC2 client is $ 0.0116 per hour. Total per month (assuming no data written and using no storage) is 252.288 USD. A price of a template solution for Hyperledger Fabric consortium on AKS relies on the service it uses. As for AKS, prices vary depending on a region – East US is assumed for calculation. Microsoft Azure Pricing Calculator [29] shows monthly cost estimation (assuming 1 cluster, 1 VM instance of type B2S) of 103.37 USD. For IBM, the pricing plan varies depending on the country or region and does not include the tax. IBM introduced a pricing model, where the costs are defined for each virtual core abbreviated as virtual processor core (VPC). With IBM Blockchain Platform, there is only standard plan available with a monthly price for all the tooling you need to build, operate, and grow your blockchain solution. For a total costs estimate three variables must be included: IBM Blockchain Platform components, IBM Cloud Kubernetes cluster and persistent storage required for the blockchain ledger. Total per month (assuming free Kubernetes cluster and no storage) is 211.70 USD.

Consentio reproduction of write throughput experiment was performed at IBM Blockchain Platform. The physical infrastructure was much less advanced than the original one, which was based on FastFabric [12] framework. As claimed by Agarwal et al., Consentio was proved to be a scalable system, since the lower TPS measurements resulting from the differences in physical infrastructure, using lower values (key space with a size of 11 280), the framework used, the presence of Endorser, were showing the same linear trend. As claimed in the Consentio paper [1], the proposed CMS is replicable and indeed scalable.

6 CONCLUSIONS

This study aimed at blockchain systems hosted by cloud vendors in the form of BaaS. The currently available solutions offered by the most appreciated cloud providers were analysed. The most promising services have been tested for the real deployment of the CMS for which the idea is excerpted from Agarwal et al. article [1]. Implementing the CMS atop BaaS 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. During the research we decided to analyse the expected minimal monthly costs for the BaaS. The key result of this empirical study is recreating the Consentio blockchain network was only possible with one of the tested BaaS platforms, the IBM Blockchain Platform. Consentio reproduction of write throughput experiment was performed at IBM Blockchain Platform. The physical infrastructure was much less advanced than the original one, which was based on FastFabric framework. As claimed by Agarwal et al., Consentio was proved to be a scalable system, since the lower TPS measurements resulting from the differences in physical infrastructure, using lower values (key space with a size of 11 280), the framework used, the presence of Endorser, were showing the same linear trend. 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.

REFERENCES

[1] Agarwal, R. R.—Kumar, D.—Golab, L.—Keshav, S.: Consentio: Managing Consent to Data Access Using Permissioned Blockchains. 2019, arXiv: 1910.07110.

[2] Amazon Web Services. AWS Blockchain Templates. 2020, `https://aws.amazon.com/blockchain/templates`, [access 18.04.2020].

[3] Amazon Web Services. AWS Marketplace Blockchain Category. 2020, `https://aws.amazon.com/marketplace/`, [access 18.04.2020].

[4] Business Wire. AWS Announces General Availability of Amazon Managed Blockchain. 2019, `https://www.businesswire.com/news/home/20190430006195/en`, [access 18.04.2020].

[5] Chen, L.—Xu, L.—Shah, N.—Gao, Z.—Lu, Y.—Shi, W.: On Security Analysis of Proof-of-Elapsed-Time (PoET). In: Spirakis, P., Tsigas, P. (Eds.): Stabilization, Safety, and Security of Distributed Systems (SSS 2017). Springer, Cham, Lecture

Notes in Computer Science, Vol. 10616, 2017, pp. 282–297, doi: 10.1007/978-3-319-69084-1_19.

[6] ConsenSys. Kaleido Enterprise Blockchain SaaS. 2018, https://aws.amazon.com/marketplace/pp/ConsenSys-Kaleido-Enterprise-Blockchain-SaaS/B07CSLDS7R, [access 17.04.2020].

[7] Dias, J. P.—Ferreira, H. S.—Martins, Â.: A Blockchain-Based Scheme for Access Control in E-Health Scenarios. In: Madureira, A., Abraham, A., Gandhi, N., Silva, C., Antunes, M. (Eds.): Proceedings of the Tenth International Conference on Soft Computing and Pattern Recognition (SoCPaR 2018). Springer, Cham, Advances in Intelligent Systems and Computing, Vol. 942, 2020, pp. 238–247, doi: 10.1007/978-3-030-17065-3_24.

[8] Dymek, D.—Grabowski, M.—Paliwoda-Pękosz, G.: Blockchain Awareness Among Computer Science Students: A Preliminary Study. In: Pańkowska, M., Sandkuhl, K. (Eds.): Perspectives in Business Informatics Research (BIR 2019). Springer, Cham, Lecture Notes in Business Information Processing, Vol. 365, 2019, pp. 30–43, doi: 10.1007/978-3-030-31143-8_3.

[9] 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. White Paper. 2016, https://www.healthit.gov/sites/default/files/5-56-onc_blockchainchallenge_mitwhitepaper.pdf.

[10] Google. Ethereum by Google Click to Deploy. 2018, https://console.cloud.google.com/marketplace/details/click-to-deploy-images/ethereum, [access 18.04.2020].

[11] Google. Hyperledger Fabric and Composer by Google Click to Deploy. 2018, https://console.cloud.google.com/marketplace/details/click-to-deploy-images/hyperledger-fabric-and-composer, [access 18.04.2020].

[12] Gorenflo, C.—Lee, S.—Golab, L.—Keshav, S.: FastFabric: Scaling Hyperledger Fabric to 20 000 Transactions per Second. 2019 IEEE International Conference on Blockchain and Cryptocurrency (ICBC), 2019, pp. 455–463, doi: 10.1109/bloc.2019.8751452.

[13] Grodzicka, H.: Solutions for Scalability and Security for Private Permissionless Blockchain. Master's Thesis, Wrocław University of Science and Technology, 2020.

[14] Gupta, S.—Malhotra, V.—Singh, S. N.: Securing IoT-Driven Remote Healthcare Data Through Blockchain. In: Kolhe, M., Tiwari, S., Trivedi, M., Mishra, K. (Eds.): Advances in Data and Information Sciences. Springer, Singapore, Lecture Notes in Networks and Systems, Vol. 94, 2020, pp. 47–56, doi: 10.1007/978-981-15-0694-9_6.

[15] Huertas, J.—Liu, H.—Robinson, S.: Eximchain: Supply Chain Finance Solutions on a Secured Public, Permissioned Blockchain Hybrid. Eximchain White Paper, 2018, https://cryptorating.eu/whitepapers/Eximchain/Whitepaper%20-%20Eximchain.pdf.

[16] Hummer. Sharding FAQ. 2017, https://github.com/ethereum/wiki/wiki/Sharding-FAQ, [access 12.12.2019].

[17] Hyperledger. Hyperledger Fabric Release 2.0 Documentation. 2020, https://hyperledger-fabric.readthedocs.io/en/release-2.0, [access 18.04.2020].

[18] IBM. Blockchain Development – IBM Developer. 2019, https://developer.ibm.com/technologies/blockchain, [access 26.04.2020].

[19] IBM. Blockchain Platform – IBM Cloud. 2020, cloud.ibm.com/catalog/services/blockchain, [access 23.04.2020].

[20] IBM Blockchain. IBM Blockchain Platform – Visual Studio Marketplace. 2018, https://marketplace.visualstudio.com/items?itemName=IBMBlockchain.ibm-blockchain-platform, [access 24.04.2020].

[21] Jesus, V.: Towards an Accountable Web of Personal Information: The Web-of-Receipts. IEEE Access, Vol. 8, 2020, pp. 25383–25394, doi: 10.1109/access.2020.2970270.

[22] Kedziora, M.—Kozlowski, P.—Jozwiak, P.: Security of Blockchain Distributed Ledger Consensus Mechanism in Context of the Sybil Attack. In: Fujita, H., Fournier-Viger, P., Ali, M., Sasaki, J. (Eds.): Trends in Artificial Intelligence Theory and Applications. Artificial Intelligence Practices (IEA/AIE 2020). Springer, Cham, Lecture Notes in Computer Science, Vol. 12144, 2020, pp. 407–418, doi: 10.1007/978-3-030-55789-8_36.

[23] Kedziora, M.—Pieprzka, D.—Jóźwiak, I.—Liu, Y.—Song, H.: Analysis of Segregated Witness Implementation for Increasing Efficiency and Security of the Bitcoin Cryptocurrency. In: Nguyen, N. T., Hoang, B. H., Huynh, C. P., Hwang, D., Trawiński, B., Vossen, G. (Eds.): Computational Collective Intelligence (ICCCI 2020). Springer, Cham, Lecture Notes in Computer Science, Vol. 12496, 2020, pp. 640–651, doi: 10.1007/978-3-030-63007-2_50.

[24] Lu, Q.—Xu, X.—Liu, Y.—Weber, I.—Zhu, L.—Zhang, W.: uBaaS: A Unified Blockchain as a Service Platform. Future Generation Computer Systems, Vol. 101, 2019, pp. 564–575, doi: 10.1016/j.future.2019.05.051.

[25] Microsoft. Kaleido Enterprise Blockchain SaaS by ConsenSys. 2019, https://azuremarketplace.microsoft.com/en-us/marketplace/apps/consensys.kaleido, [access 17.04.2020].

[26] Microsoft. Microsoft Azure Marketplace. 2019, https://azuremarketplace.microsoft.com, [access 17.04.2020].

[27] Microsoft. MultiChain on Azure by MultiChain. 2019, https://azuremarketplace.microsoft.com/en-us/marketplace/apps/coin-sciences-ltd.multichain-on-azure, [access 17.04.2020].

[28] Microsoft. Quorum by Enterprise Ethereum Alliance. 2019, https://azuremarketplace.microsoft.com/en-us/marketplace/apps/enterprise-ethereum-alliance.quorum-demo, [access 18.04.2020].

[29] Microsoft. Microsoft Azure Pricing Calculator. 2020, https://azure.microsoft.com/en-us/pricing/calculator, [access 13.05.2020].

[30] Muratov, F.—Lebedev, A.—Iushkevich, N.—Nasrulin, B.—Takemiya, M.: YAC: BFT Consensus Algorithm for Blockchain. 2018, arXiv: 1809.00554.

[31] Onik, M. M. H.—Miraz, M. H.: Performance Analytical Comparison of Blockchain-as-a-Service (BaaS) Platforms. In: Miraz, M., Excell, P., Ware, A.,

Soomro, S., Ali, M. (Eds.): Emerging Technologies in Computing (iCETiC 2019). Springer, Cham, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Vol. 285, 2019, pp. 3–18, doi: 10.1007/978-3-030-23943-5_1.

[32] Oracle. Blockchain Platform – Oracle Cloud. 2020, `https://www.oracle.com/application-development/cloud-services/blockchain-platform`, [access 17.04.2020].

[33] Oracle and/or its affiliates. Oracle Cloud Marketplace. 2020, `https://cloudmarketplace.oracle.com/marketplace`, [access 17.04.2020].

[34] R3. Corda. 2020, `https://aws.amazon.com/marketplace/pp/B07MXHSR6P`, [access 17.04.2020].

[35] Rebelo, J.: Blockchain Technology Impact on Supply Chain Management. Master's Thesis, NOVA – School of Business and Economics, 2019.

[36] Reyna, A.—Martín, C.—Chen, J.—Soler, E.—Díaz, M.: On Blockchain and Its Integration with IoT. Challenges and Opportunities. Future Generation Computer Systems, Vol. 88, 2018, pp. 173–190, doi: 10.1016/j.future.2018.05.046.

[37] Trojanowska, N.—Kedziora, M.—Hanif, M.—Song, H.: Secure Decentralized Application Development of Blockchain-Based Games. 2020 IEEE 39$^{\text{th}}$ International Performance Computing and Communications Conference (IPCCC), 2020, pp. 1–8, doi: 10.1109/ipccc50635.2020.9391556.

[38] Wright, D.—Smith, D.—Bala, R.—Gill, B.: Gartner Magic Quadrant for Cloud Infrastructure as a Service, Worldwide. Gartner, 2019.

[39] Wu, J.—Dong, M.—Ota, K.—Li, J.—Yang, W.: Application-Aware Consensus Management for Software-Defined Intelligent Blockchain in IoT. IEEE Network, Vol. 34, 2020, No. 1, pp. 69–75, doi: 10.1109/mnet.001.1900179.

[40] 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, Vol. 6, 2018, No. 2, pp. 1594–1605, doi: 10.1109/jiot.2018.2847705.

[41] Zyskind, G.—Nathan, O.—Pentland, A. S.: Decentralizing Privacy: Using Blockchain to Protect Personal Data. 2015 IEEE Security and Privacy Workshops, 2015, pp. 180–184, doi: 10.1109/spw.2015.27.

Hanna Grodzicka is Professional Software Engineer. As one of top students, she graduated from the Wroclaw University of Science and Technology receiving Engineer's and Master's degrees in computer science in 2020. Her research areas of interests encompass blockchain and software quality.

Michal Kedziora received his Ph.D. degree in computer science from the Wroclaw University of Science and Technology, Wroclaw, Poland, in 2014, and his M.Sc. and Engineer degree in computer security from the University of Technology in Wroclaw, Poland, in 2006. In 2017 he finished PostDoc at the University of Wollongong, Australia. He was working as Visiting Researcher at University of Technology Sydney, Australia (2019) and the Embry-Riddle Aeronautical University, Daytona Beach, FL, USA (2020). In 2014, he joined the Department of Applied Informatics at the Faculty of Computer Science and Management, University of Science and Technology in Wroclaw, Poland, where he is currently Assistant Professor and the Head of the Cyber Security MiniLab as part of the Department Laboratory. His research interests include computer security, cryptography, digital forensics, blockchain and artificial intelligence algoritms. He is also experienced IT Security Professional and Digital Forensics Investigator with many years working in public and private sector, including being Law Enforcement Officer and Digital Forensic Court Expert. He worked in Orange Labs (France Telecom R & D Center) as Chief Security Architect and as Senior IT Security Expert at IBM. His IT Security knowledge and skills are backed up by the following certifications: CISSP, Certified Ethical Hacker, CompTIA Security+, EnCE. He is a member of IEEE and The Polish Information Processing Society.

Lech Madeyski is Associate Professor and Deputy Head for Research at the Department of Applied Informatics, Wroclaw University of Science and Technology, Poland. He has been Visiting Researcher at the Keele University, Brunel University London, and Visiting Professor at the Blekinge Institute of Technology. His research is focused on empirical (evidence-based) software engineering, data science in software engineering, robust statistical methods, reproducible research, software quality, mutation testing, agile methods. He is a co-founder of e-Informatica Software Engineering Journal. He published, e.g., in IEEE Transactions on Software Engineering, Empirical Software Engineering, Information and Software Technology, and authored "Test-Driven Development: An Empirical Evaluation of Agile Practice" book by Springer. He serves as steering committee member, program co-chair, workshops/special sessions/tracks co-chair, and PC member of international conferences in software engineering.