Blockchain-based Smart Contracts: A Systematic Mapping Study of Academic Research (2018)

Maher Alharby, Amjad Aldweesh, Aad van Moorsel
School of Computing, Newcastle University
Newcastle upon Tyne, UK
Email: {m.w.r.alharby2, a.y.a.aldweesh2, aad.vanmoorsel}@ncl.ac.uk

***Abstract*—Blockchain based smart contracts are computer programs that encode an agreement between non-trusting participants. Smart contracts are executed on a blockchain system if specified conditions are met, without the need of a trusted third party. Blockchains and smart contracts have received increasing and booming attention in recent years, also in academic circles. We carry out a systematic mapping study of all peer-reviewed technology-oriented research in smart contracts. Our interest is twofold, namely to provide a survey of the scientific literature and to identify academic research trends and uptake. We only focus on peer-reviewed scientific publications, in order to identify how academic researchers have taken up smart contract technologies and established scientific outputs. We obtained all research papers from the main scientific databases, and using the systematic mapping method arrived at 188 relevant papers. We classified these papers into six categories, namely, security, privacy, software engineering, application, performance & scalability and other smart contract related topics. We found that the majority of the papers falls into the applications (about 64%) and software engineering (21%) categories. Compared to our 2017 survey [1], we observe that the number of relevant articles has increased about eightfold and shifted considerably towards applications of smart contracts.**

***Index Terms*—Blockchain, Smart contract, Systematic mapping.**

I. Introduction

Centralized systems rely on a trusted third party (e.g., banks) to allow non-trusting participants to communicate and send financial transactions between each other. Relying on a trusted third party, however, could result in security and privacy issues as well as high transactional costs. Blockchain technology aims to address this by allowing non-trusting participants to reach a consensus on their transactions and communications without the involvement of a trusted third party. Blockchain can be thought of as a distributed database that maintains the history of all transactions that have ever occurred in the blockchain network. Blockchain is the underlying innovation behind the first distributed electronic payment system, Bitcoin. Blockchain has evolved to support a number of decentralized applications beyond financial applications. Many of these applications rely on the execution of smart contracts on top of the blockchain.

A smart contract is a computer program that encodes the agreement between non-trusting participants and is executed based on some pre-defined rules [2]. A smart contract is deployed or executed on blockchain systems as part of a blockchain transaction. Miners, special type of participants in the blockchain network, are responsible for deploying new contracts and executing existing ones. Miners get paid for this job based on the computational costs required to execute the contracts. The most popular platforms that support deploying and executing smart contracts are Ethereum and Hyperledger Fabric.

The aim of this paper is to identify and to classify all peer-reviewed research that has been conducted on smart contract technology. Importantly, we do not attempt to include all the latest developments in technical, financial or political issues that were communicated through other channels, particularly the Internet. We also are interested in longitudinal (year after year) aspects of academic contributions to smart contracts, to document and analyze the growth in research outputs and compare the emphasis on specific topics across years (specifically compared to our 2017 survey [1]). In addition, this paper aims at finding open issues in smart contracts that need to be tackled by future research. To achieve this, we decided to follow the systematic mapping study approach proposed in [3] to look for relevant papers in the main scientific databases and to generate a classification map. The generated map helps to better understand the topics of interest as well as identify gaps for future work.

The structure of this paper is as follows. Section II provides background information about blockchain and smart contract technologies. In Section III, we present the methodology used to conduct the systematic mapping study, including the definition of the research questions. Section IV illustrates the results of searching and screening for relevant papers as well as the results for classifying all research papers and comparing to previous years. In Section V, we discuss and answer the research questions. Section VI concludes the paper.

II. Background

This section presents some background information about blockchain and smart contracts technologies. In addition, it discusses some blockchain platforms for supporting the development of smart contracts.

A. Blockchain Technology

A blockchain is a distributed database that maintains the history of all transactions that have ever occurred in the blockchain network. All network nodes have a copy of this database making it replicated and backed up. One advantage

of a blockchain is that it allows non-trusting participants to communicate and exchange assets in a secure manner without the need of a trusted third party. As the name indicated, a blockchain is a series of blocks connected to each other through a cryptographic hash. Each block references the previous block by storing its cryptographic hash, resulting in a chain of blocks. In addition to the previous block's hash, a block contains a set of transactions that, once accepted and appended to the blockchain cannot be updated or deleted. As a result, both integrity and double-spending problems are mitigated.

Cryptocurrencies such as Bitcoin [4] represent the first generation of systems that utilize the blockchain technology. Cryptocurrencies are digital currencies that combine the advantages of a peer-to-peer network with cryptographic techniques. Bitcoin is the most popular cryptocurrency with a market capitalization around $112B as of July 2018. All transactions performed in the Bitcoin network are verified by a set of nodes in the network. Those nodes are called miners and any node can be a miner. Miners are responsible for validating transactions by checking the transaction signatures as well as account balances. Following the process of validation, miners create a block after solving a computationally intensive mathematical puzzle called Proof-of-Work. The created block is sent to the network where others validate the correctness and add it to their blockchain copy. Although Bitcoin uses a form of smart contract, building complex applications on top of Bitcoin blockchain is non-trivial.

Ethereum is the second most popular cryptocurrency, distinguishing itself through its ability to deploy and run complex distributed applications on top of the blockchain. Ethereum achieves this using Turing complete smart contracts, which will be discussed in detail later and run within the Ethereum Virtual Machine. Using Ethereum, distributed applications can be developed using different high-level programming languages such as Solidity.

B. Smart Contract Overview

A smart contract is a computer program that runs on the blockchain. A smart contract can be considered as a trusted third party between non-trusting participants. Smart contracts consist of a contract storage, a balance, and program code. It can be created and made available for use by any node in the network, simply through posting a transaction to the blockchain. Smart contract program code is fixed and cannot be updated once included in the blockchain.

Smart contracts are run by a network of miners who are responsible for maintaining the blockchain. Miners reach consensus on the execution outcome of the smart contract and accordingly update the blockchain. Once deployed, each smart contract is assigned to 160-bit address and is executed whenever a transaction is created using this address. During the execution of the smart contract its storage might be updated (i.e., reading from or writing to the storage). In addition, a smart contract can exchange cryptocurrency between users. Moreover, a smart contract can invoke and create another smart contract by posting a message, which is not recorded in the blockchain. This message is used by smart contracts either for creating a new smart contract or for calling functions in other smart contracts.

C. Smart Contract Platforms

Smart contracts can be developed and deployed in different blockchain platforms (e.g., Ethereum). Different platforms offer different features for developing smart contracts. In this section, we will only focus on three platforms, which are Bitcoin, Ethereum and Hyperledger Fabric.

Bitcoin [4] is a permissionless blockchain platform that supports cryptocurrency transactions. Bitcoin uses a stack-based bytecode scripting language that is very limited in terms of compute capability [5]. Bitcoin scripting language cannot support the creation of complex smart contracts that contain rich logic. For instance, writing contracts that support loops or withdrawal limits is not feasible in Bitcoin due to the limitations of its scripting language [2].

Like Bitcoin, Ethereum [2], [6] is a permissionless blockchain and cryptocurrency. In addition to the ability to transfer currency it supports building and running complex applications based on smart contracts on the blockchain. Ethereums native currency is called Ether. The basic unit of the Ethereum system is account. In Ethereum there are two types of accounts: an externally owned accounts and a contract account. The former is controlled by the corresponding private key of the owner and keeps a balance. It also can be used for transactions to transfer money or to invoke a smart contract. The later is controlled by the smart contract code logic and it has balance, storage, and state. At the heart of Ethereum is the Ethereum virtual machine, which executes smart contracts. The source code of a smart contract is compiled into a bytecode form which can be interpreted by the Ethereum virtual machine. Each node of Ethereum runs the same instructions to facilitate the execution of smart contracts and to maintain the blockchain consensus. Ethereum smart contracts can be developed in many Turing-complete languages such as Solidity.

Bitcoin and Ethereum have scalability challenges, in that at most tens to hundreds transaction per second can be processed. Hyperledger Fabric [7] is a permissioned blockchain that overcomes these challenges. Hyperedger Fabric employs a traditional Byzantine fault-tolerant consensus protocol, instead of the Proof-of-Work protocol employed in permissionless blockchains. Hyperledger's smart contract technology is called chaincode. It consists of the code that is deployed and executed on the blockchain, the state database (key/value store) and the mining (endorsement) policies.

III. Research methodology

We chose the systematic mapping study proposed by [3] as our research methodology in order to explore the current research related to smart contracts technology. A systematic mapping approach allows us to identify and classify research topics relevant to smart contracts. It also helps us to identify

research gaps for future research. The systematic mapping study is divided into five steps, namely, defining research questions, conducting the search, searching for relevant papers, keywording using abstract and data extraction.

A. Defining research questions

The first step in a systematic mapping study is to define the research questions to be answered by the study. For our study, we have defined four questions, which are as follows:

RQ1. What are the existing research areas in smart contracts?
RQ2. How does smart contract research evolve year on year in terms of the number and type of publications?
RQ3. What existing applications are there for smart contracts?
RQ4. What are the research gaps?

B. Conducting the search

The second step is to search and gather all research papers related to smart contracts based on a specific search term. We chose the term 'smart contract' for this study as the main search keyword. Having identified the keyword for the searching task, we selected five different scientific databases to carry out our search. The selected databases are ACM Digital Library, Springer, IEEE Explore, Scopus and ScienceDirect. We only focus on gathering peer-reviewed research papers published in journals, conferences, symposiums, workshops and books.

C. Screening for relevant papers

The third step is to exclude all research papers that are irrelevant to our research questions. To accomplish this step, we followed the screening approach proposed in [8]. First, we attempted to remove irrelevant research papers based on their titles. If we could not manage to decide on the relevancy of a paper based on their title, we would run through a second step by evaluating the abstract of that paper. In addition to title and abstract based exclusion, we also relied on some exclusion criteria to remove some papers. We removed papers without English text, without full text available, with no critical contributions such as popular articles, newsletters or grey literature. Moreover, we removed duplicate papers and non-technology based papers.

D. Keywording using abstracts

The fourth step is to classify all the relevant research papers based on the keyword approach proposed in [8]. We went through the abstract of all papers in order to associate crucial keywords and the key contribution. The purpose of doing so is to classify all research papers under different categories. In some cases where it was difficult to classify a paper using its abstract, we skimmed the paper quickly to make a proper decision about its category.

E. Data extraction and mapping process

The last step is to collect all information needed to answer the research questions of our study. We collected various data items embracing the main goal and contribution from each research papers.

IV. Study Results

This section discusses the results of the systematic mapping study that we conducted on smart contracts. We first discuss the results of searching and screening for relevant papers. Then, we discuss the results of the classification process.

A. Searching and Screening Results

The first two steps are searching and screening. During the searching phase, we searched for all scientific papers in various scientific databases using a specific keyword 'smart contract'. We gathered 617 papers in total, on 12 June 2018. During the screening phase, we first excluded duplicate papers. We ended up with 407 unique papers. After that, we went through the title and the abstract for all the 407 papers in order to exclude irrelevant papers. We managed to exclude 219 irrelevant papers (about 54% of all papers). There are three reasons why we had a high number of excluded papers. First, many papers were irrelevant to our study, since our focus was to explore smart contracts from a technical perspective. For instance, many papers discussed the topic from an economic or legal point of view. Another reason is that some excluded papers were about cryptocurrencies or blockchain in general (as opposed to smart contracts), which do not contribute to our research questions. The last reason is that some papers were excluded as they only discuss gray literature about smart contracts or discuss the possibility of applying them to different domains such as Internet of Thing, without providing any technical contribution. Therefore, we ultimately included 188 papers in our systematic mapping study.

B. Classification Results

By applying the Keywording technique, we classified the papers into six categories, namely, security, privacy, software engineering, application, performance & scalability and other smart contract related topics. Security relates to bugs or vulnerabilities that an adversary might utilize to launch an attack in smart contract systems. Privacy includes issues related to disclosing contracts information to unauthorized people. Software engineering refers to any work related to the software development of smart contracts. Application refers to the utilization of smart contracts to address issues in different domains such as Internet of Thing (IoT). Performance & scalability refers to the ability of smart contract systems to deliver a reasonable response time as well as to sustain performance when the number of contracts is increasing.

Figure 1 shows the percentage of scientific papers in each of the six categories. It is clear that most papers are smart contract applications, accounting for 64% of all the papers. The second most common category is software engineering, with 21% of the papers. Security category dominates 6% of

the papers. 3% and 2% of the papers fall into performance & scalability and privacy categories respectively. It is worth noting that there are some papers (4% of all papers) fall into other smart contract related topics.

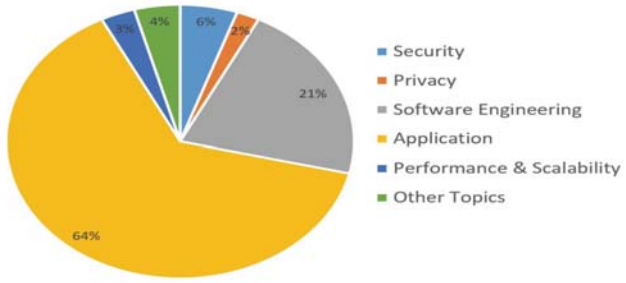

Fig. 1. The percentage of scientific papers in each category.

Figure 2 shows the total number of publications per year and the number of publications in each category per year. Research on smart contracts started in 2015 with the emergence of the Ethereum blockchain, the most common platform for smart contracts. We only found 2 papers published in 2015, one of them is an application paper while the other one belongs to software engineering category. In 2016, the number of scientific papers increased significantly by 22 papers and since then it further increased dramatically. Nearly a hundred papers more were published in 2017. In the first half of 2018, already 70 papers found their way into the above-mentioned research databases. Notably, in the 2017 survey [1] we considered only 24 papers to be relevant, which is about a factor eight less than in this paper. We hope to conduct the same review again in 2019, and one can imagine the increase in papers we will see then.

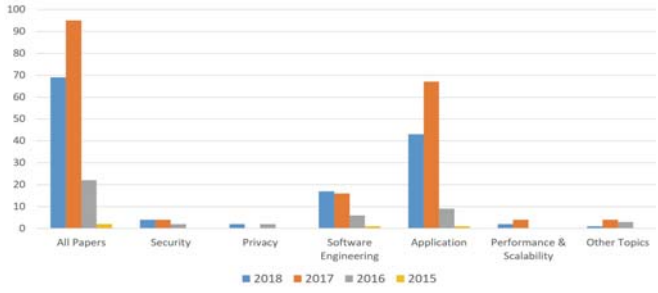

Fig. 2. Number of publications in each category per each year.

Security. We found 10 papers that identify or propose solutions to bugs and vulnerabilities in smart contract systems. Some papers present detection tools (e.g., Oyente [9] and ReGuard [10]) that can be used by developers to identify common security bugs. Some papers propose assurance methods and machine learning approaches to detect security risks and fraud. Furthermore, one of the papers identifies three types of criminal activities that can be carried out in smart contract systems [11]. Another paper proposes an adaptive incentive mechanism for smart contract systems to defend against denial of service attacks [12]. In addition, Atzei et al. [13] surveyed several vulnerabilities in Ethereum smart contracts and built a taxonomy for such vulnerabilities.

Privacy. We only found 4 papers that extend blockchain-based smart contract platforms to support privacy and confidentiality. Hawk [14], for instance, is a tool that allows smart contract developers to build privacy-preserving contracts. The rest of the papers focus on extending Hyperledger Fabric to support private data, proposing encrypted data feeds for contracts and proposing a system to ensure the confidentiality of contracts.

Software Engineering. 40 papers fall into software engineering for smart contracts, covering a wide range of topics. About 20% of the papers focus on verification and validation techniques (e.g., formal modeling techniques [15]) for smart contracts in order to assure that smart contracts are functioning as intended as well as error-free. Six papers are about proposing new platforms (such as Smartdemap [16]) and languages (e.g., Simplicity [17]) for developing smart contracts. Three papers utilize analysis techniques to inspect the code of smart contracts. Moreover, there are three papers that propose solutions to blockchain immutability features by allowing the modification and termination of already deployed smart contracts. Two papers focus on optimizing the code of smart contracts by identifying and solving programming patterns with high execution costs.

The rest of the papers focuses on automating the process of developing contracts, building code classifiers and parsers for contracts, human-centered design of contracts, designing templates for developing contracts, profiling smart contract interactions and identifying common pitfalls in developing safe contracts. Furthermore, there are some papers that propose frameworks for developing secure contracts, proposing a compression method to reuse previously deployed contracts and proposing the integration of a semantic legal layer with the blockchain to support legal contracts.

Applications. We found 120 application-based papers (about 64% of all papers). We classified these applications into several topics, namely, Internet of Thing (IoT), cloud computing, financial, data, healthcare, access control & authentication and other applications.

Internet of Thing (IoT) Applications: Internet of Thing (IoT) refers to physical devices and appliances connected via the Internet. We found 18 papers that apply blockchain-based smart contract technology to IoT. Three applications utilize smart contracts to build an access control system for IoT. Four applications utilize smart contracts to overcome security and privacy issues in the IoT. The rest of IoT-based applications utilize smart contracts for the management of IoT devices, electronic business, data management, data trading, data exchange and data storage.

Cloud Computing Application: We found 8 applications that utilize blockchain-based smart contract to overcome various technical issues in cloud computing. These applications address the issues of verifiability of outsourced computation [18], data auditing, resource management of cloud datacentres,

negotiation and agreement establishment, data accountability, trust, access control and service level agreement (SLA) monitoring.

Financial Applications: We found several applications that utilize blockchain-based smart contracts for financial purposes such as payment and loan. The identified financial applications embrace fair payment systems, privacy-preserving incentive mechanisms, a smart will, taxation-based payment, car insurance, private and concurrent payment channel network, concert tickets and protocols for data trading and the management of study loan repayment.

Data Applications: These types of applications utilize blockchain-based smart contract to manage and secure general data and information. These applications include data sharing, data management, data indexing, data integrity check, and data provenance and accountability.

Healthcare Applications: We found three blockchain-based smart contract applications in the healthcare domain. These applications are a secure remote patient monitoring system, an access control framework for electronic health records and trustless medical data sharing among different cloud providers.

Access Control & Authentication Applications: These types of smart contract applications target approaches to user authentication and management of access right policies. Several smart contract based access control systems have been proposed in different domains such as healthcare [19], IoT [20] and cloud computing [21]. With regard to authentication, there are different proposed applications such as a secure mutual authentication for industry 4.0, an enhancement of TLS handshake authentication and a distributed and secure user authentication.

The rest of the applications covers a wide range of different topics including e-voting, supply chain management, intelligent systems (e.g., intelligent transportation systems), smart grid systems, energy-based applications, resource management, reliable decision making, digital rights management, human resource systems and 2 phase commit protocol for distributed consensus protocols [22]. Furthermore, other applications include volunteer time record systems, QoS-aware service composition, logistics management, trustless intermediation in marketplaces, assessment organization service, Business Process Management (BPM) Systems and decentralized applications (DAPPs).

Performance & Scalability. We found 6 papers that fall into performance and scalability topics in smart contract systems. Some papers propose benchmarking frameworks (e.g., Blockbench [23]) for analyzing and monitoring the performance of blockchain-based smart contracts. To overcome scalability issues in smart contract systems, some papers propose solutions to execute smart contracts in parallel instead of sequentially.

Other smart contract related topics. We found 8 papers that fall into other smart contract related topics such as consensus protocols and incentive mechanisms for smart contracts. Some papers propose new secure consensus protocols for smart contracts and identify issues in existing protocols. Other papers focus on incentive mechanisms, system operations and system design for smart contracts.

V. Discussion

This section discusses the study results and answers the research questions that we defined in Section III.

RQ1: What are the existing research areas in smart contracts?

The study divides the research topics on smart contracts into six categories, namely, security, privacy, software engineering, application, performance & scalability and other smart contract related topics. The majority of the research (about 64%) falls into the application category, followed by software engineering (21%). The applications cover a wide range of domains such IoT, cloud computing, finance, healthcare, access control, authentication and others. Most of these applications are to address technical issues (e.g., security issues) or to get rid of the trusted third parties in existing applications. In the software engineering category, most of the papers utilize analysis techniques for validation and verification purposes or to propose new platforms and languages for developing secure smart contracts. For the performance & scalability category, the papers either propose frameworks for performance analysis or scalable solutions for the execution of contracts. In the security category, most papers focus on identifying and tackling security bugs and vulnerabilities. In the privacy category, most papers focus on the confidentiality of information in smart contract systems.

RQ2. How does smart contract research evolve year on year in terms of the number and type of publications?

The number of publications on smart contracts has increased dramatically since the second half of 2017. The number of publications has increased by about 700% (164 new papers have been published) since the systematic study [1] we conducted in May 2017. Both application and software engineering categories have experienced high number of publications. This indicates that smart contract systems widespread very vastly, especially in terms of smart contract applications and development.

RQ3: What existing applications are there for smart contracts?

Smart contract applications cover any solution that utilize smart contract technology to overcome the issues in existing systems or any smart contract based tool. We identified 120 application-based papers that make use of smart contract technology in existing systems such as IoT. These applications include cloud-based applications, healthcare-based application, financial applications, data applications, access control & authentication based applications, e-voting, smart grid systems, digital right management, intelligent systems and other decentralized applications. In addition, we identified several smart contracts tools that can aid during the process of developing smart contracts, identify security issues,

or provide confidentiality for smart contract information.

RQ4: What are the research gaps?

From this study, we are able to identify at least two research gaps in smart contract research. The methodologies used to identify these gaps are by observing issues or limitations from the papers included in this study and by relying on our knowledge in smart contract topic.

The first one is the relative lack of research on the scalability issues of blockchain-based smart contract systems. In current systems, smart contracts are executed in sequence which leads to low throughput. Although we found two papers exploring parallel execution of contracts, their proposed solutions are high-level ideas and still not proven to be working properly in smart contract systems.

A second area is the relative lack of research on performance benchmarking of smart contract execution. We have not found research that investigates and proposes benchmarking approaches to assess whether the incentive awarded to the miners for smart contract execution is proportional to the computational costs required to execute the contract. If the incentive is not aligned with the computational costs, this could result in security attacks as well as poor incentive and cost models [24], which impact the reliability of smart contract systems.

VI. Conclusion

In this paper, we conducted a systematic mapping study in order to identify and to classify all peer-reviewed research papers on smart contracts. The aim of doing so is to understand current research areas on smart contracts, to identify research gaps for future work and to identify academics trends in uptake and emphasis. We extracted 188 papers from five different scientific databases and classified these papers into six categories, namely, security, privacy, software engineering, application, performance & scalability and other smart contract related topics. The number of papers not only increased with a factor of eight compared to our survey of 2017 [1], but we also found an enormous increase in papers presenting application, now accounting for the majority of papers (about 64%).

References

[1] M. Alharby and A. van Moorsel, "Blockchain-based smart contracts: A systematic mapping study," *arXiv preprint arXiv:1710.06372*, 2017.

[2] V. Buterin, "A next-generation smart contract and decentralized application platform.," Available online at: https://github.com/ethereum/wiki/wiki/White-Paper/ [Accessed 19/07/2018].

[3] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic mapping studies in software engineering.," in *EASE*, vol. 8, pp. 68–77, 2008.

[4] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008.

[5] A. Lewis, "A gentle introduction to smart contracts.," Available online at:https://bitsonblocks.net/2016/02/01/a-gentle-introduction-to-smart-contracts/ [Accessed 19/07/2018].

[6] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum Project Yellow Paper*, vol. 151, 2014.

[7] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, *et al.*, "Hyperledger fabric: a distributed operating system for permissioned blockchains," in *Proceedings of the Thirteenth EuroSys Conference*, p. 30, ACM, 2018.

[8] J. Yli-Huumo, D. Ko, S. Choi, S. Park, and K. Smolander, "Where is current research on blockchain technology?a systematic review," *PloS one*, vol. 11, no. 10, p. e0163477, 2016.

[9] L. Luu, D.-H. Chu, H. Olickel, P. Saxena, and A. Hobor, "Making smart contracts smarter," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 254–269, ACM, 2016.

[10] I. Nikolic, A. Kolluri, I. Sergey, P. Saxena, and A. Hobor, "Finding the greedy, prodigal, and suicidal contracts at scale," *arXiv preprint arXiv:1802.06038*, 2018.

[11] A. Juels, A. Kosba, and E. Shi, "The ring of gyges: Investigating the future of criminal smart contracts," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 283–295, ACM, 2016.

[12] T. Chen, X. Li, Y. Wang, J. Chen, Z. Li, X. Luo, M. H. Au, and X. Zhang, "An adaptive gas cost mechanism for ethereum to defend against under-priced dos attacks," in *International Conference on Information Security Practice and Experience*, pp. 3–24, Springer, 2017.

[13] N. Atzei, M. Bartoletti, and T. Cimoli, "A survey of attacks on ethereum smart contracts (sok)," in *Principles of Security and Trust*, pp. 164–186, Springer, 2017.

[14] A. Kosba, A. Miller, E. Shi, Z. Wen, and C. Papamanthou, "Hawk: The blockchain model of cryptography and privacy-preserving smart contracts," in *2016 IEEE symposium on security and privacy (SP)*, pp. 839–858, IEEE, 2016.

[15] K. Bhargavan, A. Delignat-Lavaud, C. Fournet, A. Gollamudi, G. Gonthier, N. Kobeissi, N. Kulatova, A. Rastogi, T. Sibut-Pinote, N. Swamy, *et al.*, "Formal verification of smart contracts: Short paper," in *Proceedings of the 2016 ACM Workshop on Programming Languages and Analysis for Security*, pp. 91–96, ACM, 2016.

[16] M. Knecht and B. Stiller, "Smartdemap: A smart contract deployment and management platform," in *IFIP International Conference on Autonomous Infrastructure, Management and Security*, pp. 159–164, Springer, 2017.

[17] R. O'Connor, "Simplicity: a new language for blockchains," in *Proceedings of the 2017 Workshop on Programming Languages and Analysis for Security*, pp. 107–120, ACM, 2017.

[18] C. Dong, Y. Wang, A. Aldweesh, P. McCorry, and A. van Moorsel, "Betrayal, distrust, and rationality: Smart counter-collusion contracts for verifiable cloud computing," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 211–227, ACM, 2017.

[19] G. G. Dagher, J. Mohler, M. Milojkovic, and P. B. Marella, "Ancile: Privacy-preserving framework for access control and interoperability of electronic health records using blockchain technology," *Sustainable Cities and Society*, vol. 39, pp. 283–297, 2018.

[20] H. F. Atlam, A. Alenezi, R. J. Walters, G. B. Wills, and J. Daniel, "Developing an adaptive risk-based access control model for the internet of things," in *Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2017 IEEE International Conference on*, pp. 655–661, IEEE, 2017.

[21] I. Sukhodolskiy and S. Zapechnikov, "A blockchain-based access control system for cloud storage," in *Young Researchers in Electrical and Electronic Engineering (EIConRus), 2018 IEEE Conference of Russian*, pp. 1575–1578, IEEE, 2018.

[22] P. Ezhilchelvan, A. Aldweesh, and A. van Moorsel, "Non-blocking two phase commit using blockchain," in *Proceedings of the 1st Workshop on Cryptocurrencies and Blockchains for Distributed Systems*, pp. 36–41, ACM, 2018.

[23] T. T. A. Dinh, J. Wang, G. Chen, R. Liu, B. C. Ooi, and K.-L. Tan, "Blockbench: A framework for analyzing private blockchains," in *Proceedings of the 2017 ACM International Conference on Management of Data*, pp. 1085–1100, ACM, 2017.

[24] M. Alharby and A. van Moorsel, "The impact of profit uncertainty on miner decisions in blockchain systems," *Electronic Notes in Theoretical Computer Science*, vol. 340, pp. 151–167, 2018.