Blockchain-based Distributed Marketplace

Oliver R. Kabi and Virginia N. L. Franqueira

University of Derby, Cyber Security Research Group
College of Engineering & Technology
DE22 1GB, Derby, United Kingdom
o.kabi1@unimail.derby.ac.uk; v.franqueira@derby.ac.uk

Abstract. Developments in Blockchain technology have enabled the creation of smart contracts; i.e., self-executing code that is stored and executed on the Blockchain. This has led to the creation of distributed, decentralised applications, along with frameworks for developing and deploying them easily. This paper describes a proof-of-concept system that implements a distributed online marketplace using the Ethereum framework, where buyers and sellers can engage in e-commerce transactions without the need of a large central entity coordinating the process. The performance of the system was measured in terms of cost of use through the concept of 'gas usage'. It was determined that such costs are significantly less than that of Amazon and eBay for high volume users. The findings generally support the ability to use Ethereum to create a distributed on-chain market, however, there are still areas that require further research and development.

Keywords: Blockchain · Smart Contract · Ethereum · E-Commerce · Distributed Systems

1 Introduction

In order to exchange physical goods online, internet users utilise third party marketplaces to mitigate some of the associated risks of trading with unknown Internet users and to also facilitate the processing of the financial transaction. Buyers and sellers expect that the third party marketplace will act as a trusted intermediary and provide the service of connecting and protecting peers wishing to exchange goods, in exchange for a fee that is likely a percentage cut of the transaction. Despite these protections, reported instances of non-payment and non-delivery crimes amounted to over $138 million in losses for victims in the USA in 2016 [14].

The use of extremely large and reputable marketplaces, such as eBay and Amazon, offer better protection to users from retail fraud than small, unknown alternative marketplaces, however this in turn creates powerful centralised corporations with vast quantities of personal information about users, which can be processed and sold [19]. Recent events have brought to light the issue of data privacy, creating wide spread concern regarding the collection and use of user data by large corporations [11]. Alongside this are concerns regarding the security of our sensitive financial information online. Again, recent events such as the Equifax data breach have highlighted the risks of sharing financial information, even with large, previously trusted financial corporations [7]. In order to exchange currency online users must provide sensitive personal and banking information, exposing their data to the risk of a data breach and thus exposing the user to the risk of identity theft. Large marketplaces pose a double threat of information security to users, as users must expose both their personal information, that is valuable for the study and manipulation of populations through big data analysis, as well as their banking information which could be used for identity theft, should it be compromised.

Recent developments in Blockchain technology have enabled the creation of Smart Contracts, self-executing contracts that are stored and executed on the Blockchain nodes, allowing secure decentralised applications to be developed. Buterin et al. [8] suggest many applications for smart contracts, including on-chain decentralised marketplaces, however they provides no details on such an application. By using Smart Contracts to create an on-chain decentralised marketplace, the requirement of a large central entity to co-ordinate the marketplace functions are removed, which in turn provides a method to remove data aggregation and user exposure to the threats posed.

The contribution of the paper is threefold. First, it reports on the design and implementation of blockchain technology for a distributed physical goods marketplace application, in order to enable trading of goods without the requirement of a trusted third party market operator using the Ethereum framework. Second, it evaluates the developed proof-of-concept marketplace system in terms of performance, based on *gas* used for computation by the smart contracts on the blockchain, and scalability. Third, it discusses issues and insights gained which give directions to future research & development.

The reminder of the paper is organised as follows. Section 2 provides background on traditional online marketplaces, related work and the Ethereum framework. Section 3 enumerates requirements for online marketplaces and presents the architecture of the proof-of-concept system. Section 4 elaborates on the aspects of the developed system and the testing performed. Section 5 evaluates the system in terms of performance and scalability, while Section 6 provides discussion based on the work done, issues faced and suggestions for future work. Finally, Section 7 draws conclusions.

2 Background

This section reviews (in Section 2.1) current online commerce and the technologies employed by existing centralised marketplaces to facilitate the online exchange of goods and services , despite the prevalence of fraud and other risks faced by online peers. Following this related work is discussed to provide a basis for our design. It also reviews the Ethereum framework (in Section 2.3), which is used to implement the decentralised marketplace proof-of-concept system.

2.1 Online Commerce

While online commerce provides convenience to consumers, online retail fraud is one of the top three crime types reported in the United States of America with losses of $138 million for victims in the US in 2016 [14]. These losses are for non-payment or non-delivery of goods exchanged between online peers, which highlights the risks faced by both buyer and sellers in online marketplaces.

To combat the increased opportunity for fraud that exists on the trustless web, a number of legal and technological solutions are employed. A number of these solutions will now be detailed.

Centralised marketplaces. Centralised marketplaces act as an intermediary between buyers and sellers, helping to facilitate the exchange of goods for currency. A peer is able to list an item for sale, which another peer is then able to purchase in exchange for currency via a payment provider like PayPal or a credit card provider like VISA. Consumers may mitigate the risk of fraud by using large trusted online retailers or marketplaces such as Amazon or eBay. Even when making purchases from third parties via these large marketplaces, the marketplace acts as an intermediary between the consumer and seller, mediating any disputes that may arise from the transaction.

Escrow services. A more traditional form of intermediary that can be utilised are escrow services. In this process, the buyer and seller agree upon a mutually trusted third party, that will be responsible for the funds until the conditions of the transaction, as agreed upon by all parties, have been met. Should any dispute arise the third party will provide arbitration and attempt to resolve the dispute. However as shown by Grazioli and Jarvenpaa [16], most users are unable to distinguish fraudulent websites, leaving consumers open to fraudulent online escrow services.

Reputation Systems. Many online marketplaces, including Amazon and eBay, provide reputation systems which provide a metric for users to judge one another and determine the risk of partaking in an exchange with another particular user. These trust and reputation systems are being used successfully by online commercial applications to promote good behaviour by users [18]. Resnick and Zeckhauser [25] report that reputation profiles used on eBay were predictive of future performance and they found that users participate in reputation systems the majority of the time, despite the incentives to free ride.

2.2 Related Work

Subramanian [27] provides an analysis of decentralised marketplaces on the blockchain based on the current theory and literature. He identifies advantages of using blockchain-based marketplaces to include, faster transaction times, reduced costs and increased privacy and security for users. He also provides an analysis of the decentralisation possibility for various product and service categories, including physical products which he states may only achieve partial decentralisation due to the complexities of providing decentralised B2B support, accounting, payment and reputation systems.

Notheisen, Cholewa, and Shanmugam [23] implemented a real-world asset marketplace with a private blockchain, using the ethereum framework to create a digital motor vehicle asset register. The system provides automated transfer of ownership along with immutable records of vehicle history. The study also introduced methods to reduce the risk arising from the immutability of blockchain transactions. By providing abort mechanisms, both buyer and seller may disengage from the transaction before final confirmation and exchange of funds and property. The application shifts the centralisation onto a single government authority to provide verification of vehicle information and condition, however it's benefits provide a system that requires less work to participate in and maintain compared to existing vehicle registration methods.

Mobius [20] provides protocols for smart markets however the aim is more specifically for providing a marketplace for autonomous agents to trade data and micro services, specifically with the world of IoT in mind.

Nasonov et al. [21] propose a big data platform that companies can use to sell, exchange and process Big Data sets. Such an application would provide a valuable mechanism for companies to improve their business organisation through the application of knowledge extracted from such data sets.

Eskandari et al. [13] studied the possibility of using smart contracts to implement a derivatives market that would remove the need for an intermediary broker. They highlight gaps in the current infrastructure, stopping the development of secure and autonomous derivatives markets, particularly the lack of decentralised data feeds, as this may lead to potential vulnerabilities should centralised feeds become compromised. They also highlight issues raised by new development concepts such as gas usage, and limiting the gas usage during computation.

2.3 Ethereum

Ethereum extends the application of the blockchain technology used by Bitcoin to provide a Turing complete scripting language. Scripts can be committed to the blockchain via transactions, indefinitely making them publicly accessible. This enables the ability to encode arbitrary state transition functions, and as such the ability to create decentralised blockchain-based applications, otherwise known as Dapps.

Smart Contracts. The Turing-complete scripting language incorporated into Ethereum allows for the creation of smart contracts as envisioned by Szabo [29]. Scripts can be included within transactions as in Bitcoin, however contracts also have their own addresses that can send and receive transactions, allowing parts of the contract to be executed upon receiving a transaction. Smart contracts are able to communicate with one another through messages allowing for complex interactions to be developed such as Decentralised Autonomous Organisations [9].

Ether and Gas. Gas, which is directly exchangeable for Ether (ETH) at a consistent real price, is the unit used to fuel the computations of byte-code within the EVM and storage of data on the blockchain. The use of a resource to fuel contracts protects from malicious and infinitely looping code from being executed on the chain without the cost of computation being paid for.

Contracts and transactions have a fixed start price to pay the miner for his computational power. Each specific computation also has predefined gas costs, such as additions, subtractions, memory stores and retrievals, as such the total cost of computation can be estimated on compilation of the code. However as ethereum is still being developed, changes could cause the gas cost of a contract function to change.

3 System Design

To effectively investigate the research question, an adaptation of the Design Science Research (DSR) methodology set out by Vaishnavi and Kuechler [30] will be used. This process involves investigating a known problem through the design and creation of an artefact, which can then be evaluated through discussion and reflection as a potential solution to the original problem.

The intent is to provide a system for the trading of physical goods similar to eBay, meaning that both the physical characteristics and the value of the goods will be extremely varied. As such the automatic verification of physical exchange of goods is a problem beyond the scope of what is currently achievable by this paper. In order to resolve disputes during exchange, such as non-delivery, misrepresented goods or fraudulent claims, some form of centralisation will be required, such as an escrow and arbitration process.

We used abstract story-like descriptions using non-technical language to define the required behaviour of the system, inspired by Agile software development [24]. Table 1 contains a list of user stories from the perspective of the various user types of the application. The users are subdivided into "Buyers", those wishing to purchase items, and "Sellers", those wishing to list and sell items. These requirements are identified using a sequential numbering *Rn* to provide easy referencing.

The application is designed to have a back-end consisting of two Solidity smart contracts hosted on the Ethereum public blockchain, along with a HTML/CSS/JS front end UI application. The web application will use the web3 API to interact with the smart contracts. Figure 1 provides a diagram of the systems architecture.

3.1 Back-end

The back-end will consist of two smart contracts written in Solidity, the Marketplace Agent contract and the Escrow Agent contract. Each will act as a sort of repository for the business data they are required to administer and store, with

Table 1. User Stories encompassing the requirements of the blockchain-based distributed marketpkace system.

ID	User Story
R1	As a seller I am able to list items for sale with a listing name and price in Ether.
R2	As the seller of a purchased listing, I am able to approve the linked escrow for the aforementioned listing.
R3	As the seller of a purchased listing, my ethereum account is funded with the proceeds of a sale once the buyer has approved/finalised the linked escrow, once the listed goods have exchanged possession.
R4	As a buyer I am able to browse existing listing for sale.
R5	As a buyer I am able to purchase an available listing.
R6	As a buyer I am able to fund an escrow for the purchase of a listing.
R7	As the buyer of a listing, I am able to approve the linked escrow for the aforementioned listing.
R8	As the buyer or seller of a listing, I am able to request and receive impartial mediation of an escrow in the event of a issue with the exchange of goods.

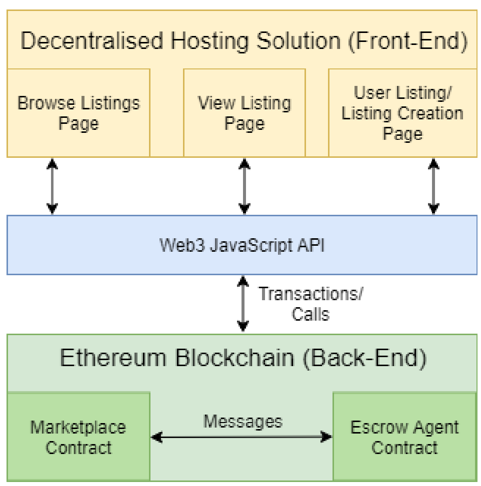

Fig. 1. System Architecture

their implementation based on the pattern described by Hitchins [17]. These contracts will now be further described.

Marketplace Contract. The marketplace contract will provide the business logic which orchestrates the process of listing, browsing and purchasing of items. It will also handle the logistical information exchange between a buyer and a seller required to enable the logistics of a physical exchange.

Escrow Agent Contract. The escrow contract will provide the business logic to reduce and mitigate risk for both buyers and sellers in the exchange of physical goods over the internet. This logic will form an escrow contract between the buyer and seller that is paid out automatically in the event that both the buyer and seller approve the escrow. Logic for disputing the escrow will also be provided, in which case the escrow can be paid out by an arbitrator to either the buyer or the seller.

3.2 Front-end

While the smart contracts provide all the business logic of a marketplace, using them via an API would not be user friendly and as such a JavaScript web application will be implemented to provide a front-end that can be accessed via a web browser.

The front-end of the marketplace will consist of a typical web application. This will be comprised of a JavaScript application that utilises the Web3 API to interact with both the marketplace and escrow contracts along with HTML pages to provide a graphical user interface.

4 System Implementation

The requirements R1-R8 (Table 1) were processed into a Kanban style to-do list, allowing the development of the application in a Behaviour-Driven Development (BDD) fashion [26] This allowed the developer to better react to an evolving understanding of the development processes and methods within the Ethereum framework. The development environment consisted of the components and tools enumerated in Table 2.

The architecture of the smart contract implementation is described by the UML diagram of Figure 2. The source code for the components which compose the architecture of the system shown in Figure 1, i.e. the Marketplace and Escrow Agent contracts, the "browser listing page", "view listing page", "user listing/listing creation page", and "user orders page" can be found on GitHub: `https://github.com/Howserr/onchain-market`.

5 System Evaluation

The proof-of-concept system has been tested and evaluated in terms of white-box testing, black-box testing [22] and cost of use for users. Test scripts and full results are also available on GitHub.

White-box Testing. A suite of 74 unit tests were created during development to provide white box test coverage of both smart contracts, MarketplaceAgent and EscrowAgent. These tests were organised as BDD structures using the Mocha architecture. Naming of the groups and individual tests followed the behavioural language style of "given, when, then". These tests were created to run using the Truffle testing framework which runs against a local in-memory blockchain, i.e., Ganache.

Black-box Testing. A set of 8 test cases were created to meet the requirements R1-R8; all of the test cases were successful except the script for R8, the

Table 2. Tools used for the implementation of the proof-of-concept system

Software	Version	Purpose
Web3	1.0	Provides a JavaScript API for Ethereum blockchains by implementing the Ethereum JSON RPC API; it allows front-end web applications to interact with blockchains.
Ganache	1.1.0	Ethereum blockchain JavaScript implementation which runs in-memory; it removes the need for a local blockchain client for local testing and provides additional tools to aid in development of smart contracts.
Truffle	4.1.7	Part of Ethereum development framework which provides a compilation, testing and deployment pipeline for Dapp developers.
Mocha	5..0	Included as part of Truffle in order to provide a unit testing framework for Solidity contract code/
MetaMask	4.5.5	Extension for the Chrome web browser that enables interaction with the Ethereum blockchain, as required for the use of Dapps.
NodeJS	8.11.1	Provides a development framework for Truffle and enables use of NPM, the NodeJS package manager, for the installation of JavaScript dependencies

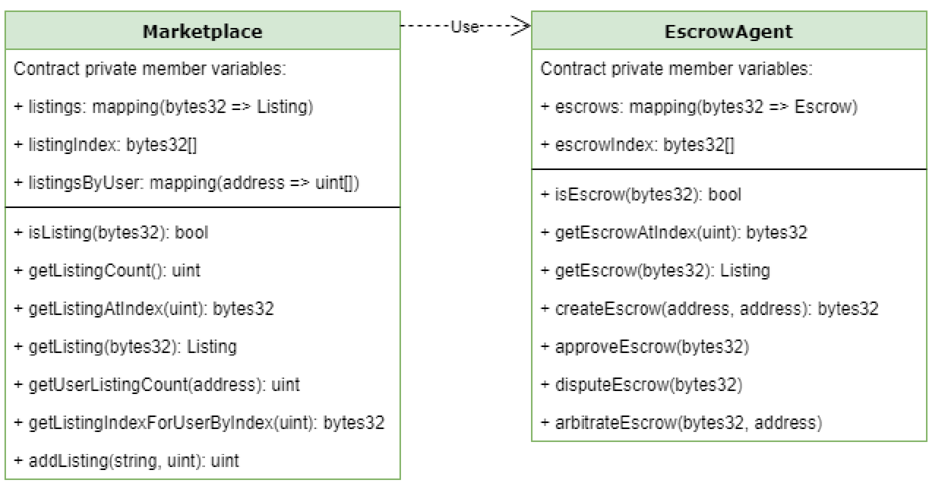

Fig. 2. Smart Contracts UML Diagram

ability for an arbitrator to provide arbitration of a disputed listing/escrow. This test failed because the front-end functionality for arbitrators was not implemented due to security issues created by the single arbitrator solution of our artefact. This problem, along with potential solutions, is discussed further in Section 6.

Performance Testing. This section presents the results of performance testing the implemented artefact in terms of the costs of use. The quantity of gas used by each action that requires sending a transaction to the blockchain was recorded using Ganache. In order to calculate a cost based on the gas used, the exchange price of gas was determined to be 1 gwei per unit. This exchange price is based on the current *Gas Price Standard* at ETH Gas Station [1]. The costs of use for the artefact are displayed in Table 3.

It should be noted that the execution of Solidity code is deterministic and the gas used is calculated as the sum of the gas used by the EVM opcodes executed. As such measurements of gas usage were not repeated.

The costs for selling on Amazon and eBay were calculated based on the pricing models detailed on their seller information pages [3, 2].The calculations were based on an item in the Consumer Electronics category, the second most preferable category of goods to buy online [4]. In both cases we used the cheapest fee structure available, those targeted at sellers running online stores, however the subscription cost is not included in the calculation. For Amazon the *consumer electronics* referral fee percentage of 8% (minimum $1.00) was used and consequently it would be a non-media item so no closing fee is charged. As such, the cost to sell an item on Amazon given the stated assumptions, is *8% of the listing price*. The cost to sell an item on eBay given the same assumptions is 10% of the total value of sale. The results of these calculations are displayed in Table 4.

Table 3. Combined buyer and seller use costs of listing and selling an item in Gas, ETH and USD (rounded up to 2 decimal places).

Behaviour	Gas Cost	ETH Cost	USD Cost
Add Listing	163192	0.0001632	$0.07
Purchase Listing	237384	0.0002374	$0.10
First Approval (seller)	45524	0.0000455	$0.02
Second Approval (buyer)	37050	0.0000370	$0.02
Total Cost	483150	0.004831	$0.21

Table 4. The costs of selling items of specific prices on popular online marketplaces in USD.

Item Price	Amazon Cost	eBay Cost
$10	$1.00	$1.00
$50	$4.00	$5.00
$100	$8.00	$10.00
$1000	$80.00	$100.00

Comparison. By comparing Tables 3 & 4 it can be seen that the combined cost of both a buyer and seller for selling an item using the artefact will be significantly lower than the costs of selling even a $10 item on either Amazon or eBay. As the costs of usage for the artefact are based on the gas used rather than a marketplace commission, the costs would not scale based on the price of the item being exchanged as it would on Amazon or eBay.

6 Discussion

This section discusses issues faced during implementation of the artefact, and suggests future work.

6.1 Escrow Arbitration

Our initial design intent was to use a similar system to that employed by existing marketplaces, in which administrative members of the marketplace community would take turns to fulfil the role of arbitrator when required. During the development phase it became obvious that the planned method of arbitration was fundamentally flawed when used within a decentralised system as it creates centralisation and a requirement of trust in a closed group of administrators. As such this requirement was not fulfilled and instead we suggest it as a subject for further study. Insights into the problem based on the current literature will now be provided.

Group Escrow. The use of group escrow protocols in order to facilitate the exchange of physical goods using cryptocurrencies has been studied previously [15]. A protocol using group escrow techniques could be developed in order to solve the requirement of trust in a single arbitrator. A number of arbitrators would be selected from a pool of registered arbitrators who would then individually review the available information pertaining to the exchange of goods and vote on the party to which the balance of the escrow should be awarded to. This then presents the problem of finding consensus between the arbitrators, a problem which the literature provides many potential avenues for solution.

Proof-of-Stake Consensus The Casper Proof-of-Stake protocol being developed as a replacement for Proof-of-Work by the Ethereum Foundation [10] provides potential solutions for encouraging good peer behaviour within a consensus system. Aspects of this protocol could be applied to group arbitration such as by requiring would-be arbitrators to stake Ether. By doing this, the stake of malicious actors can be slashed, in order to create a disincentive to such behaviour, while arbitrators that form the majority could be rewarded with a share of a fee charged as part of the arbitration process, when it is requested. The shared fee would be collected from the balance of the escrow.

Such an arbitration protocol would reduce the potential impact of malicious actors within the centralised process. It could be utilised as a separate arbitration service for many forms of transaction within applications on the decentralised web.

6.2 Logistics Integration

During an online exchange of physical goods the logistics of the physical exchange are likely to involve a logistics carrier due to the potential distances. Currently the system has no functionality to help peers organise this crucial part of the exchange of goods, or to help arbitrators understand the current state of the logistic process of the exchange. By introducing the ability for courier logs to be recorded against an escrow, the state of the exchange could be better tracked providing arbitrators with rich information to better perform their role, reducing the risk of participation for good peers. Logistical updates could be made via an interaction with the smart contract from the logistics provider, which would of course require their co-operation. However, such functionality would require authorised access such that logistic logs on the escrow could not be fraudulently created by anyone but the logistics provider; else the logs would no longer be dependable for arbitration. In order to provide authorisation to update logistics logs, logistic providers would need to have an Ethereum account verified and approved for access to the specific escrow. This creates the requirements for a process to verify that a given Ethereum account belongs to a specific and legitimate logistics provider. The integration of logistical elements into a distributed on-chain marketplace provides an opportunity for further work.

6.3 Reputation System

The introduction of an eBay style reputation system would help sellers and buyers to better identify safe, low risk peers to exchange goods with [18]. An average of a users feedback scores could be displayed as part of their profile to help peers better judge the risk of potential exchanges.

An alternative option would be to use a reputation system protocol designed for peer-to-peer networks. Swamynathan et al. [28] suggest such a system to address user collusion and short-lived online identities, the main causes of erroneous and misleading values in reputation systems. A similar reputation system that is blockchain based was introduced by Dennis and Owen [12], however no implementation currently exists for practical use.

6.4 Decentralised Front-end Hosting

The developed implementation is a locally deployed proof of concept artefact, however any real marketplace would be required to be publicly accessible. The back-end of the application, the smart contracts, would be hosted on the blockchain, however the front-end is a standard web application and so requires a web hosting solution. The simplest solution would be to use a standard web hosting service, however this would centralise the front-end and drastically reduce the points of failure required to make the application inaccessible. Decentralised web hosting options are under development, such as the IPFS peer-to-peer hypermedia protocol [6] or the ethereum foundation developed Swarm [5], a peer-to-peer storage and content distribution platform. However at the current time, while both are available to use, development of the systems is still ongoing and as such neither would be suitable for production environment use. Works to study the deployment of a Dapp front-end to a decentralised hosting platform should be undertaken.

7 Conclusion

To evaluate the applicability of an on-chain marketplace for physical goods, first the current blockchain literature was reviewed and then an proof-of-concept artefact was designed. The artefact was implemented to meet the basic requirements of a physical goods marketplace, captured in 8 requirements (R1–R8). It was then successfully tested against the requirements using white box and black box testing to verify the software. Performance measurements were taken regarding the gas usage of various behaviours of the marketplace, which were analysed against figures from existing centralised marketplaces. Our successful acceptance testing and performance testing of the artefact provides a demonstration of the Ethereum framework's testing and quality assurance capabilities – a key requirement for production applications. Discussion of the shortcomings of the generated artefact were presented, such as the vulnerabilities of a single escrow arbitrator and the difficulties regarding integrating logistics information and logs. Ultimately, it was concluded that an on-chain marketplace is indeed feasible, however, further study and development of the technology is required before a production implementation becomes practical.

References

1. ETH Gas Station, https://ethgasstation.info/, last accessed: 01/05/2018
2. How much does it cost to sell on eBay?, https://sellercentre.ebay.co.uk/private/what-fees-youll-pay, [Online] Accessed: 01/05/2018
3. Sell on Amazon - Pricing and Fees, https://services.amazon.co.uk/services/sell-online/pricing.html, [Online] Accessed: 01/05/2018
4. Total Retail 2017, https://www.pwc.com/gx/en/industries/assets/total-retail-2017.pdf, accessed: 01/05/2018
5. ethersphere/swarm (Dec 2017), https://github.com/ethersphere/swarm
6. Benet, J.: IPFS-content addressed, versioned, P2P file system. arXiv preprint arXiv:1407.3561 (2014)
7. Berghel, H.: Equifax and the Latest Round of Identity Theft Roulette. Computer **50**(12), 72–76 (2017)
8. Buterin, V.: A Next-Generation Smart Contract and Decentralized Application Platform (2013), https://github.com/ethereum/wiki/wiki/White-Paper, [Online] Last accessed 02/05/2018; document constantly updated by different people.
9. Buterin, V.: Bootstrapping A Decentralized Autonomous Corporation: Part I. Bitcoin Magazine (2013), https://bitcoinmagazine.com/articles/bootstrapping-a-decentralized-autonomous-corporation-part-i-1379644274/
10. Buterin, V., Griffith, V.: Casper the Friendly Finality Gadget. arXiv preprint arXiv:1710.09437 (2017)

11. Colangelo, G., Maggiolino, M.: Data Accumulation and the Privacy-Antitrust Interface: Insights from the Facebook Case for the EU and the US (2018), working Paper No. 31/2018; available: https://ssrn.com/abstract=3125490
12. Dennis, R., Owen, G.: Rep on the block: A next generation reputation system based on the blockchain. In: In Proc. of the 10th Int. Conference for Internet Technology and Secured Transactions (ICITST). pp. 131–138. IEEE (2015)
13. Eskandari, S., Clark, J., Sundaresan, V., Adham, M.: On the feasibility of decentralized derivatives markets. In: International Conference on Financial Cryptography and Data Security. pp. 553–567. Springer (2017)
14. FBI: 2016 Internet Crime Report (2016), `https://pdf.ic3.gov/2016_IC3Report.pdf`, [Online] Last accessed 02/05/2018.
15. Goldfeder, S., Bonneau, J., Gennaro, R., Narayanan, A.: Escrow protocols for cryptocurrencies: How to buy physical goods using bitcoin. In: International Conference on Financial Cryptography and Data Security. pp. 321–339. Springer (2017)
16. Grazioli, S., Jarvenpaa, S.L.: Perils of Internet fraud: An empirical investigation of deception and trust with experienced Internet consumers. IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans **30**(4), 395–410 (2000)
17. Hitchens, R.: Solidity CRUD - Part 1 (2017), `https://medium.com/@robhitchens/solidity-crud-part-1-824ffa69509a`, [Online] Accessed: 01/05/2018
18. Jøsang, A., Ismail, R., Boyd, C.: A survey of trust and reputation systems for online service provision. Decision support systems **43**(2), 618–644 (2007)
19. Leber, J.: Going Head to Head with Google and Facebook, Amazonś Next Big Business Could Be Advertising. MIT Technology Review (2014)
20. Mochi Inc.: Mobius: A Universal Protocol Suite for the Blockchain Ecosystem and Real World Data (Nov 2017), `https://mobius.network/mobius_white_paper.pdf`, [Online] Last accessed 02/05/2018.
21. Nasonov, D., Visheratin, A.A., Boukhanovsky, A.: Blockchain-based transaction integrity in distributed big data marketplace. In: International Conference on Computational Science. pp. 569–577. Springer (2018)
22. Nidhra, S., Dondeti, J.: Black box and white box testing techniques-a literature review. International Journal of Embedded Systems and Applications (IJESA) **2**(2), 29–50 (2012)
23. Notheisen, B., Cholewa, J.B., Shanmugam, A.P.: Trading real-world assets on blockchain. Business & Information Systems Engineering pp. 1–16 (2017)
24. Ramesh, B., Cao, L., Baskerville, R.: Agile requirements engineering practices and challenges: an empirical study. Information Systems Journal **20**(5), 449–480
25. Resnick, P., Zeckhauser, R.: Trust among strangers in Internet transactions: Empirical analysis of eBay's reputation system. In: The Economics of the Internet and E-commerce, pp. 127–157. Emerald Group Publishing Limited (2002)
26. Smart, J.F.: BDD in Action: Behavior-driven development for the whole software lifecycle. Manning Publications (2014)
27. Subramanian, H.: Decentralized blockchain-based electronic marketplaces. Communications of the ACM **61**(1), 78–84 (2017)
28. Swamynathan, G., Almeroth, K.C., Zhao, B.Y.: The design of a reliable reputation system. Electronic Commerce Research **10**(3-4), 239–270 (2010)
29. Szabo, N.: Formalizing and securing relationships on public networks. First Monday **2**(9) (1997). https://doi.org/10.5210/fm.v2i9.548
30. Vaishnavi, V., Kuechler, W.: Design Science Research in Information Systems (2004), `http://desrist.org/desrist/content/design-science-research-in-information-systems.pdf`