A Blockchain-based System for Online Consumer Reviews

K. Salah, A. Alfalasi, M. Alfalasi
Department of Electrical and Computer Engineering
Khalifa University of Science and Technology, Abu Dhabi, UAE
Email: khaled.salah@ku.ac.ae

***Abstract*—Today's online review systems (used largely by hospitality services, internet retailers, government services, etc.) are highly centralized and subject to tampering and manipulation. Implementing a decentralized, trustworthy, unbiased and transparent review system is a major challenge. In this paper, we present a solution that utilizes Ethereum Blockchain, Smart Contracts, and IPFS to provide a secure, transparent and trusted platform for an online review system with high integrity and resiliency. We discuss key aspects related to architectural design, interactions between system components, algorithms and logic flow. We also show how we implemented and tested the overall system functionalities. Furthermore, we provide vulnerability analysis of the smart contract code. The full smart contract code was made publicly available at Github.**

Keywords – Online Consumer Reviews, Blockchain, Smart Contracts, Ethereum, IPFS

I. INTRODUCTION

Online reviews have recently seen enormous usage and growth. Popular online review systems are being used by Booking.com, Tripadvisor, Airbnb, Yelp, as well as online shopping stores as Amazon, Alibaba, and eBay. The credibility of these reviews has been (and is being) questionable, as the reviews can be modified, manipulated, or deleted by the central agency [1-6]. Recently, as was reported in [6], an Australian hotel chain was fined $2.2 million for deleting negative reviews on the popular TripAdvisor website. Other than TripAdvisor, there are similar cases of vote manipulation, omitted or deleted negative reviews, and other dubious methods that multiple review websites such as Yelp and Airbnb have been found employing [7,8]. This manipulation can come in many forms, such as fake reviews, omission of negative reviews, and promotion of advertised reviews. This stems from the fact that all review systems, to date, are mostly centralized and subject to compromise, hacking, and tampering of its data by insider or outsiders, and also subject to failure and outside attacks.

In this paper, we propose a blockchain-based review system that provides a decentralized trusted online reviews, with the ability of tracking and tracing reviews to the original reviewers. Storing reviews on the Blockchain ensures that this review will never be altered, manipulated or deleted by any entity. Our solution combines the use of Ethereum networks and smart contracts with the Inter-Planetary File System (IPFS). IPFS is a peer-to-peer distrusted files system used for trusted, secure, and reliable storage.

Our proposed Blockchain-based review system uses Ether (which is the cryptocurrency for Ethereum) to incentivize reviewers to provide review. Service providers have the ability to issue tokens to legitimate reviewers. Tokens will be used to validate reviewers and allow for uploading reviews. The reviewers get paid in Ether upon successful upload of a review. With the use of Ethereum smart contracts, such feature can be implemented. In short, a service provider, such as TripAdvisor, uploads the rules of conducts captured in a Smart Contract code to the public Ethereum network and deposits Ether to be dispensed to legitimate reviewers. The user will then be able to submit a review using the IPFS on a front-end application. After validating the review, the Smart Contract code will reward the user with Ether for their efforts, and the review will be stored permanently in the IPFS system, to be accessed by all with high integrity and reliability.

The main objective of this paper is to illustrate how blockchain smart contracts can be used apace with IPFS to introduce a decentralized, credible, trusted online review system. The main contributions of this paper can be summarized as follows:

- We propose a blockchain-based solution using Ethereum smart contracts and IPFS for an online review system that is highly reliable, secure, decentralized, and trusted. The system allows tokens to be issued to valid reviewers, and have them paid in Ether upon successful completion of online review.
- We present details on system overview, design, architecture, and the interactions of the various actors in the system.
- We provide a complete implementation of the Ethereum smart contract code and discuss key aspects related to the frontend and backend interfaces used by actors.
- We provide testing to show the correct system functionality, with security and vulnerability analysis for the Ethereum smart contract code.

The remainder of the paper is organized as follows. Section II gives some background information on smart contracts and blockchain. Section III summarizes the related work that exists in the literature. Section IV presents our proposed blockchain based online review solution. Finally, Section V concludes the paper.

II. BACKGROUND

In this section, we describe necessary background information related to Blockchain, Ethereum, and IPFS which are the essential components used for our proposed online review system.

978-1-7281-1878-9/19/$31.00 ©2019 IEEE

A. Blockchain

Blockchain is a public, open source digital ledger that is distributed among peers in a network. This public ledger is immutable, i.e., it cannot be altered once recorded. It holds a permanent record of transactions and interactions between the users in a decentralized network. Blockchain utilizes a shared ledger that is distributed among the users of the network. It is recorded in a sequential chain of blocks that are linked through a cryptographic hash. Each block contains the information of the transaction and the asset exchanges (i.e., ether or bitcoin) that took place between the users [9]. This technology became popular after the advent of crypto currencies, such as bitcoin, which started in the year 2009.

B. Ethereum Smart Contracts

Ethereum is one of the programmable blockchain platforms that allow anyone to build decentralized applications. The difference between Ethereum and the regular blockchain is that it is considered flexible and anyone can build robust applications easily. It is considered programmable since it allows users to create their own set of definitions instead of using predefined procedures such as bitcoin transactions. The fundamental aspect that defines the Ethereum platform is the Ethereum Virtual Machine (EVM) that can run code of varying complexities [10]. Smart Contracts are pieces of code that run on the blockchain network such as Ethereum.

C. IPFS

The Inter-Planetary File System is a decentralized file system that is connected across nodes of computers that share a common file system. It works similar to bit torrent networks; it is a Peer-to-Peer file system network where the transfer of files occurs. It is content-addressable, which means the contents of IPFS can be accessed through the use of IPFS hash addresses. Moreover, this file system is indisputable, since it works similarly to the blockchain network by having a list of nodes, and it doesn't allow any tampering of files. In addition, due to its decentralized nature, there isn't a single point of failure i.e., if one device got disconnected, the file can still be accessed [11].

III. Related work

There is number of centralized review systems that are found in the literature, with varying degrees of trustworthiness and reliability. However, the literature is lacking trusted solutions that utilize blockchain to have completely indisputable reviews in a decentralized system. As blockchain technology has just recently emerged, most systems are in early stages and are high-level in nature, and have not published any implementation information. In this section, we will summarize four blockchain-based review systems that claim the use of Ethereum smart contracts to implement a review platform.

Revain [12, 13] is a platform for reviews that uses IBM's Artificial Intelligence. It filters out and removes potentially fake and low-quality reviews, safeguarding "good" quality reviews on Ethereum blockchain, where they can no longer be tampered with or modified. In its early iterations, reviews on the platform are limited to successfully completed ICO and crowd funding crypto currency projects. However, they intend to expand into e-commerce, booking, and gaming. This system is a combination of three components: Firstly, they developed a token system using two tokens (R and RVN) to create a stable coin. Secondly, they employed IBM's AI system utilizing a neural network to scan and measure the quality of reviews, in order to filter out unsolicited reviews. Finally, an immutable blockchain is utilized to record all reviews so that they are never tampered with, modified, deleted or removed. After a review is authenticated, it becomes part of the chain which cannot be removed without breaking the chain. The two tokens have different uses in the system, and it cannot function without either one. Token R is for gathering funds during ICO, and token RVN is used internally in the system to ensure stable exchange rates. This approach enables the system to reward quality reviews, while also penalizing fake reviews and companies that negatively affect their reputation. Since quality reviews are rewarded, this acts as an incentive for users to post good-quality reviews, as they will be compensated for their time. These fees required are debited to the company of the reviewed product or service, which is already registered on the Revain platform which includes the user reward, and the platform fee. The platform fee is a fee paid to the platform as a means of monetization. In order to not exclude smaller companies, the platform fee scales with the number of reviews submitted in the last 2 weeks, up to a limit. Users are also monitored based on their reviews. If the system rejects a review too many times, the user is warned, and multiple warning could lead to the user being blocked from the system, unable to withdraw their funds. Furthermore, since it is not economically feasible to store all reviews on Ethereum blockchain, only parts of the reviews are stored by a smart contract called Review Snapshots Storage.

Lina.review [14] is a review platform that utilizes blockchain technology to store user-submitted reviews in a secure manner, maintaining the integrity of the reviews and rewarding the user. Lina is a blockchain-based platform, where individuals or companies can build their own systems on the platform for free. However, while participation is free, the system will be charged maintenance fee by the platform, based on the activities of the system, specifically the participation of merchants, advertisers and the reviewing activities. All Lina-based review systems are globally connected via the platform. Merchants, parties who wish to have their products or services reviewed, and Advertisers, businesses who wish to advertise their products on the platform, must register in the system by paying a registration fee and product listing fee. Moreover, the platform employs the use of Helpers or Experts. Experts are users who are qualified to review products and services on the system, and their reviews will be displayed separately. In order to become an Expert or a Helper, the user must either sign up in the system, providing a CV and proof of domain knowledge or be promoted from a common user to a Helper by publishing many reviews that are accepted by the public. Common users are not rewarded for their reviews unless they are promoted to Helpers. Helpers are entitled to receive a considerable amount of the advertisement revenue or registration fee. However, to maintain their helper status, they must satisfy a monthly quota of acceptable reviews, and failing to do so may lead to demotion. There is no actual way to verify that the user submitting the review has actually used the product or service that he is reviewing, and therefore, reviews can be fabricated. In order to address the issues related to the economic feasibility of storing reviews on Ethereum blockchain, Lina platform utilizes

a hybrid architecture, bridging between the public chain of Ethereum, and a "high performance, scalable private side-blockchain transaction service for scalable interactions with the LINA token" called the Lina Core, which is used to store detailed transactions. This solution aims to avoid the heavy transaction fees imposed by Ethereum for their services as well as reducing the stress on the public network.

Dentacoin [15, 16] claims to be the first blockchain-based platform aimed at trusted dental services reviews. The platform provides multiple solutions: Firstly, it provides a platform where dentists can register their dental offices in the system, to be reviewed by the public. By doing so, they can view patient feedback as well as valuable market research data for free. Additionally, patients can also register dental offices and clinics that they have visited. Secondly, patients are able to provide unfiltered reviews of their dentists, based on their own personal experiences. Additionally, to provide even more trustworthy reviews, the concept of "trusted reviews" is introduced, which differs from standard reviews in that only verified patients of the dental services can review them, whereas anybody can post a standard review. These patients are verified by the actual dentists by sending an email to them with a link to post a review. Posted reviews via that review request link are marked as trusted reviews. All of these activities are linked to Ethereum blockchain via a smart contract utilizing Dentacoins, and both dentists and patients are rewarded with this token. The dental currency, Dentacoins, will be distributed throughout the industry. Patients, Dentists, and clinics are able to register on the platform, and the platform checks the legitimacy of these registration requests based on criteria such as Facebook friends. Furthermore, in the future, the Dentacoin foundation aims to launch a blockchain-based concept for dental insurance, as well as a decentralized, highly protected healthcare database of the overall medical records of patients.

Zapit [17] is a blockchain-based review platform targeting Amazon's review system. It aims to introduce trust between Amazon's sellers, consumers, and reviewers. Sellers will be provided with legitimate reviews of their products that they can trust, and will be able to create an Amazon affiliate link to Zapit via an Affiliate ID, to generate additional profits on the sale of their products that are linked to Zapit. Consumers will be compensated for the time they invest into writing reviews, and will be provided with trusted reviews that they can look at before making their purchase. Moderators, who act as blockchain miners, will build their credibility and profit by earning tokens. This review platform can be accessed by the Zapit Website or the Browser Plug-in. The website is an online platform where all the approved reviews are published for the public users looking for genuine reviews. Sellers can use this website to advertise their products and increase their credibility. The browser plug-in will be used by users who want fast-access to legitimate reviews while browsing the product listings on Amazon. They will be able to view the specific reviews of the product they are investigating, as well as provide their own reviews via the plug-in.

A re-emerging issue in all previously discussed platforms is the storage of the approved reviews. Storing the reviews on Ethereum blockchain is very expensive, and is therefore infeasible. While some review systems adopted solutions such as creating their own, privatized blockchain or storing partial review information on the chain, we opted to utilize the distributed and decentralized file system - IPFS to store the reviews. Therefore, only the transactions are executed on the Ethereum network, while all the review data is stored on the content-addressable, peer-to-peer IPFS network.

IV. System Architecture, Design and Implementation

In this section, we propose and describe our solution for an online review system that utilizes Ethereum smart contracts and IPFS to provide trusted reviews, with incentives to reviewers. The system stores all reviews in IPFS, in order to avoid the heavy cost of storing information on the public Ethereum network. It is an autonomous way of incentivizing the writing of reviews through a service provider and reviewers. Once the review is verified, it cannot be altered, since it is stored in the IPFS, with high transparency, integrity and reliability.

A. System Overview and Design

The proposed solution focuses on using smart contracts to create a blockchain based online review system. Fig. 1 describes the system architecture showing different actors and entities that will interact with the smart contract. This solution automates the process of incentivizing reviewers for posting their reviews into the IPFS. Every user of the system has an Ethereum Account with an Ethereum address.

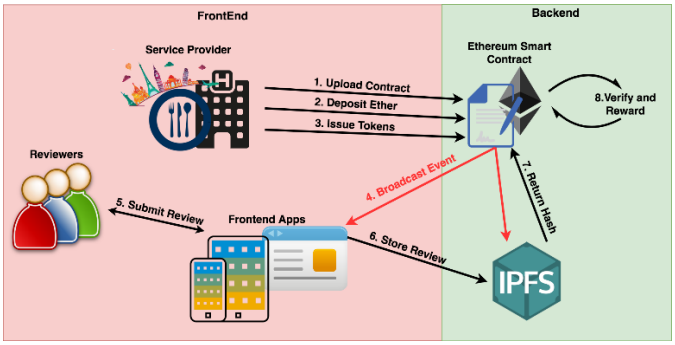

Fig. 1. System Architecture

As shown in Fig. 1, the system architecture consists of three key system components: The Service Provider, the Reviewers, and the IPFS. The participants (or actors) can be summarized as follows:

1) Service Provider: It is the entity that provides the service to the user, such as a hotel, restaurant, or an online store. The service provider will add funds (in Ether) into the smart contract to allow users to be rewarded. The service provider will also generate a token for the user to write a review.

2) Reviewers: Reviewers are the users or consumers who will submit the review through a frontend app to the IPFS. Tokens will be issued to the EA (Ethereum Addresses) of the reviewers by the service providers. The smart contract will pay Ether to EA of reviewers who completed successful reviews.

3) IPFS: IPFS will be storage system that reviews will be uploaded to. The IPFS hash of the review will be stored in the smart contract along with the reviewer's Ethereum address. The smart contract will use the token and the IPFS hash to validate the review and the legitimacy of the user.

Our solution utilizes a smart contract which is basically a code that captures the logic and algorithms for the online review dynamics and interactions. Events will be triggered in order to help track the process and cognize when a token is created or when a user has been rewarded. The smart contract contains the following primary features:

- **Modifiers:** Modifiers are functions within the smart contract to be triggered by actors. Modifiers can be restricted to be executed by a certain actors. For example, to deposit funds the modifier `onlySP()` restricts the service provider alone to add funds.
- **Events:** Events are used to track changes that occur in the smart contract and broadcast the message to all nodes in the public Ethereum network.
- **Variables:** Variables are used within the smart contract to hold state information and can be set to certain values. They have a specific data type. The main variables are `SPEthereumAddress`, `ReviewNumber`, `ReviewAward` and the `IPFS_Address` variable.

The complete smart contract code is made publically available at Github[1]. In our smart contract code, in order to authenticate the process of writing reviews, a token with a unique number will be generated for the user to write a review. It is done by using a **keccak256** hash. Once the token is created the token number will be mapped to the Ethereum address of the user. Upon token creation, an event is triggered to broadcast among the participants. Furthermore, the reward will be given once the IPFS Ethereum address will send the IPFS hash along with the token number and address of reviewer. The smart contract will then verify the token number with the address of the reviewer. The reward amount is set by the service provider. This is developed as a framework to be used with a front-end application.

B. *Implementation Details*

Our smart contract was implemented using Solidity programming language and tested using Remix IDE. The full code of our smart contract in Solidity language is made available at GitHub[1]. There are three main actors in this smart contract. Each of them is identified with an Ethereum address and call functions in the smart contract. Initially, the Service provider will create the smart contract and upload it. Then the Service Provider will deposit funds and issue tokens, in case of insufficient funds no tokens will be created. Upon token creation, a message will be broadcasted to show the Ethereum address of the user and the token following which the user can submit a review using this token. Once the review is uploaded to the IPFS server, the review token will be verified and the reviewer will be rewarded in Ether.

As shown in Fig. 2, the Service Provider will add balance to the smart contract. The `issueToken` function will create a token and it is linked to a specific Ethereum address. Once the token is created, `ReviewDone()` function will be executed after the review has been completed. The `Reward()` function will reward a reviewer with a specific amount.

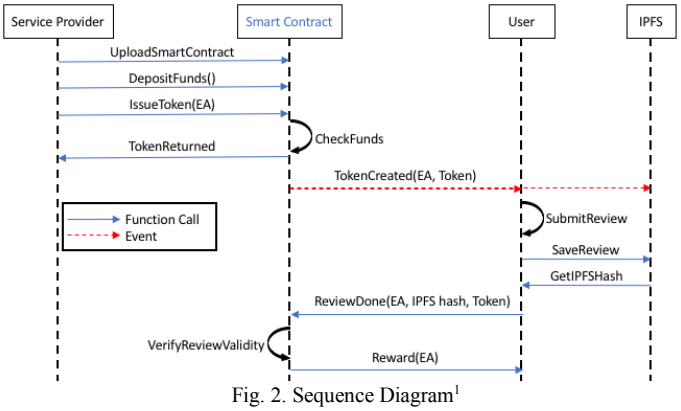

Fig. 2. Sequence Diagram[1]

Algorithm 1 shows the details of mapping, which is a key-value pair associating the Ethereum address to a specific token number which is required to allow the user to write a review. Ethereum address of the reviewer with the IPFS hash of the review are used to know the address of the user who has written the review.

Algorithm 1: Mapping for Token numbers and to store IPFS hashes

1 Create a key-value pair : For Token number that allows users to write Review [Token]
2 Create a key-value pair : To Store the IPFS Hash Of Reviewer [IPFS hash]

Algorithm 2 shows the modifiers which include the Service Provider and the IPFS server. Modifiers can be used to specify or check a condition prior to execution. In this case, we need a modifier to restrict the function access to the service provider alone. Furthermore, the IPFS server node will also have restricted access to some functions.

Algorithm 2: Modifiers for Service Provider and IPFS server

1 onlySP : Access Restricted for only Service Provider
2 onlyIPFS : Only allows IPFS server to have access

Algorithm 3 is fundamentally the constructor of the smart contract, with initializes a number of attributes and values. The `ReviewNumber` is initialized to 0 and `IPFS_Address` is set, which is an arbitrary address that can be used for the IPFS server. For testing purpose, the `RewardAmount` is set to 1 ether.

Algorithm 3: Attributes of the smart contact

Input : *Declaration of input parameters*
1 ServiceProvider(SP) creates the smart contract
2 Initialize **ReviewNumber to 0**
3 Specify Arbitrary address for **IPFS Server**
4 Declare **Rewardamount = 1 ether**

[1]The code is available at: github.com/mohammadalfalasi/BlockChain-Review-System

Algorithm 4 shows the details of `depositFunds()` function, which is a payable function, meaning that Ether can be sent to the smart contract through it. `OnlySP` modifier is used to specify that only the service provider can add funds. Since this function is payable, it requires some amount of Gas (or Ether) to be executed. If deposit is successful, it executes event `LogDep` that displays the amount sent and the funds that have been added successfully. The `showBalance()` function is used to view the current balance in the smart contract.

Algorithm 4: Smart contract function for depositing funds

Input : *Address of the sender, Amount deposited, Balance*

1 **if** *ServiceProvider(SP)* ***deposits*** *and bool ==* ***success*** **then**
2 Deposit funds into the smart contract
3 Trigger Event **LogDep**
4 Notify with a message "Funds added successfully"
5 **end**
6 **else**
7 Revert contract state and show an error.
8 **end**

Algorithm 5 illustrates the details of `IssueToken()` function, which can only be accessed by the service provider. It has a `require` statement, which checks for the condition if there is enough balance to generate the token and that the reviewer has not submitted any previous reviews (the mapping is 0 to the reviewer). Then it generates a token number and maps it to the reviewer address. An event `TokenCreated` is triggered that shows the creation of a token.

Algorithm 5: Token number generation for reviewer after checking the availability of Ether Balance

Input : $EthereumAddress(EA)$ *of the Reviewer*

1 **if** ***Current balance*** $==$ ***Available*** *and User's EA is NOT assigned any token* **then**
2 Generate Token Number
3 Map with the Reviewer's EA.
4 Broadcast Event **TokenCreated**
5 Notify with "Token created" message
6 **end**
7 **else**
8 Revert contract state and show an error.
9 **end**

Algorithm 6 presents the logic of the `reviewDone()` function. The `reviewDone()` function will verify the token number of the reviewer via mapping. The token is then removed from the user.

Finally the `reward()` function will run with the address of the reviewer so that they will receive the reward amount as illustrated in Algorithm 7. The number of reviews will also be incremented. Other steps are included in the algorithm to award a user after they submit a review to the IPFS server.

Algorithm 6: Successful completion of review followed by Token Number provision

Input : $EthereumAddress(EA)$ *of the Reviewer,* $IPFShash$, $TokenNumber$

1 **if** ***TokenNumber*** *provided to* ***Reviewer*** **then**
2 Remove token from the user
3 Execute the **Reward()** function
4 Broadcast Event **LogSent**
5 Notify with "Reviewer rewarded" message
6 Increment the **ReviewNumber**
7 **end**
8 **else**
9 Revert contract state and show an error.
10 **end**

Algorithm 7: Distribution of reward amount to Reviewers address

Input : $EthereumAddress(EA)$ *of the Reviewer*

1 Check if the current balance $>=$ Reward Amount
2 **if** *RewardAmount is sent to Reviewer's EA* **then**
3 Broadcast Event **LogSent**
4 Notify with "Reviewer rewarded" message
5 **end**
6 **else**
7 Broadcast Event **LogErr**
8 Notify with "error" message
9 **end**
10 **else**
11 Revert contract state and show an error.
12 **end**

C. Testing and Validation

In this section, we primarily focus on testing the correct interaction and functionality among system participants through the Ethereum smart contracts. Aspects related to the full system testing and implementation of the customer front-end application is left as a future work. The Remix IDE offers rich features that make it possible to test and debug smart contracts prior to deploying them. Remix IDE offers multiple Ethereum wallets, allowing to simulate a real-life scenario with an owner and multiple customers. Additionally, a debugger is also available that allows the investigation of various transactions to ensure the correct behavior of the contract.

For the purpose of testing, the Ethereum addresses of the Service provider will be `0xca35b7d915458ef540ade6068dfe2f44e8fa733c`, the IPFS server is `0x14723a09acff6d2a60dcdf7aa4aff308fddc160c` and that of the reviewer is `0x4b0897b0513fdc7c541b6d9d7e929c4e5364d2db`. Firstly, the smart contract will be uploaded into the network and is identified with its Ethereum address. Then, the smart contract is created and funds are deposited by the service provider.

Using Remix IDE, we thoroughly tested various scenarios and conditions to ensure the correct logical functionality and execution outcome from the contract, and to also assure the expected behavior of all the system components at various time points.

V. Smart Contract Vulnerability Analysis

With the advent of smart contracts, security has become one of the most important issues than before. It is important to keep the code and the related information secure, as the contract code may itself be vulnerable to attacks, given the highly distributed nature of blockhain ecosystems [18]. As an example, to spotlight the gravity of the smart contract security issues, the DAO (Decentralized Autonomous Organization) attack demonstrated the 'Reentrancy Vulnerability' that occurred within the smart contract in 2016, which led to a theft of 3.6 million ether (worth 70 million USD) [19]. Defects in programs and attackers always remain a threat to the security of the smart contract. Most commonly known smart contract vulnerabilities are Reentrancy Vulnerability, Transaction-Ordering Dependence, Call Stack Depth Vulnerability, Parity Multisig Bug and Assertion Failure. Hence, there is a firm need to secure and audit the smart contract code using automated tools.

Oyente Tool. To check the smart contract code for any possible property infringements or security vulnerabilities, we make use of Oyente tool, developed by Loi *et al* [20]. This tool analyses the smart contract code based on the technique of symbolic execution. This open source tool allows users to define the gas limit, upper bound of a number of iterations, timeout to abort the execution. Oyente also allows users to specify difficulty of the block, account state including balance and storage, and timeout values. We tested our smart contract with Oyente tool against the most known security vulnerabilities and attacks. Results reported by Oyente tool were all "False", indicating that our smart contract is secure enough and bug-free of any of the known attacks.

VI. Conclusion

In this paper, we have presented a blockchain-based solution using Ethereum smart contracts and IPFS for online review systems. We implemented Ethereum smart contracts using Solidity language, and made the code publically available on Github. We tested key functionalities of the system using Remix IDE. We demonstrated that our smart contract code is free of known vulnerabilities that can be exploited by attackers. As a future work, we are currently in the process of implementing a fully functional system with front-end decentralized applications (dApps) for reviewers and service providers.

References

[1] T. Doukoupil, "TripAdvisor accused of deleting reviews that raised red flags". CBS News. November 2nd 2017. [Online] Available: cbsnews.com/news/tripadvisor-accused-of-deleting-reviews-that-raised-red-flags/ [Accessed: 8 August 2018].

[2] S. Fenton, "TripAdvisor denies rating system is flawed, after fake restaurant tops rankings in Italy". The Independent. London. June 30th 2015 [Online] Available: independent.co.uk/life-style/gadgets-and-tech/news/tripadvisor-denies-rating-system-is-flawed-after-fake-restaurant-tops-rankings-in-italy-10354818.html. [Accessed: 8-Aug-2018].

[3] D. Lewis, "Meriton allegedly prevents guests from giving negative reviews, bribes them to improve ratings on TripAdvisor". ABC News, October 21st 2015. [Online] Available: abc.net.au/news/2015-10-21/meriton-apartments-allegedly-cheating-tripadvisor-system/6870256. [Accessed: 8-Aug-2018].

[4] BBC, "Trip Advisor rebuked over 'trust' claims by ASA". BBC News – Tech, March 8th 2012 [Online] Available: bbc.com/news/technology-16823012. [Accessed: 8-August-2018].

[5] Mail Online, "TripAdvisor removes 'reviews you can trust' slogan from its website", London: Daily Mail.co.uk, September 13th 2011. [Online] Available: dailymail.co.uk/travel/article2036846/TripAdvisor -removes-reviews-trust-slogan-website.html [Accessed: 8-Aug-2018].

[6] D. Williams, "Australian hotel chain fined $2.2 million for manipulating TripAdvisor reviews" August 1st 2018. [Online] Available: edition.cnn.com/travel/article/australia-tripadvisor-hotel-fined-trnd/index.html [Accessed: 8-Aug-2018].

[7] J. Handy, "Think Yelp is Unbiased? Think Again!!". Forbes, August 16th, 2012. [Online]. Available: forbes.com/sites/jimhandy/2012/08/16/think-yelp-is-unbiased-think-again. [Accessed: 8- Aug- 2018].

[8] "Airbnb Rating System Deceives Guests and Hosts", Airbnb Hell, March 18th, 2017. [Online]. Available: airbnbhell.com/airbnb-rating-system-deceives-guests-hosts/. [Accessed: 8- Aug- 2018].

[9] Salah, K., Rehman, M. H., Nizamuddin, N., and Al-Fuqaha, A. "Blockchain for AI: Review and Open Research Challenges," IEEE Access, Vol. 7, No. 1, December 2019, pp. 10127-10149

[10] "What is Ethereum? — Ethereum Homestead 0.1 documentation", Ethdocs.org. [Online]. Available: ethdocs.org/en/latest/introduction/what-is-ethereum.html. [Accessed: 22- Apr- 2018].

[11] J. Benet, "IPFS - Content Addressed, Versioned, P2P File System", arxiv.org, July 14th, 2014. [Online]. Available: arxiv.org/pdf/1407.3561.pdf. [Accessed: 22- Apr- 2018].

[12] "Revain: New generation feedback platform based on the blockchain technology", Revain Whitepaper. [Online]. Available: revain.org/pdf/wp/en-wp.pdf. [Accessed: 22- Apr- 2018].

[13] "Revain: Building a Trustless, Consumer Review System on the Blockchain", The Bitcoin Podcast Network. [Online]. Available: thebitcoinpodcast.com/release/revain-building-a-trustless-consumer-review-system-on-the-blockchain/. [Accessed: 22- Apr- 2018].

[14] "Lina.Review: Blockchain Based Review Platform", LinaNetwork, January 1st, 2018. [Online]. Available: lina.review/lina_whitepaper.pdf [Accessed: 22- Apr- 2018].

[15] "Dentacoin: The Blockchain Solution for the Global Dental Industry", Dentacoin. [Online]. Available: dentacoin.com/web/white-paper/Whitepaper-en1.pdf [Accessed: 22-Apr-2018].

[16] "Dentacoin Review - DCN Trusted Dental Treatments & Healthcare Insurance App?" Bitcoinexchangeguide.com. [Online]. Available: bitcoinexchangeguide.com/dentacoin/. [Accessed: 22-Apr-2018].

[17] "Understanding Zapit: The only blockchain-verified product reviews platform", Zapit, January 2018. [Online]. Available: zapit.online/Zapit_Whitepaper-v0.6.pdf [Accessed: 22-Apr-2018].

[18] Rashvand, H. F.; Salah, K.; Calero, J. M. A.; Harn, L., "Distributed security for multi-agent systems - review and applications," IET Information Security, Vol. 4, No. 4, December 2010, pp. 188-201.

[19] D. Siegel, "Understanding the DAO attack", Coindesk, June 2016. [Online]. Available: coindesk.com/ understanding-dao-hack-journalists/ [Accessed: 17-July-2018].

[20] L.Luu, D.H. Chu, H. Olickel, P. Saxena, and A. Hobor, "Making Smart Contracts Smarter", CCS 16, Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, 2016, pp 254-269.