2018 5th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2018 4th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)

Smart Collaboration Mechanism using Blockchain Technology

Rajvardhan Oak
Department of Computer Engineering
Pune Institute of Computer Technology
Pune, India
rajoak1995@gmail.com

Karanveer Singh Jhala
Department of Computer Engineering
Pune Institute of Computer Technology
Pune, India
karanjhala@gmail.com

Mrunmayee Khare
Department of Computer Engineering
Pune Institute of Computer Technology
Pune, India
khare.mrunmayee5696@gmail.com

***Abstract*—Bitcoin is a decentralized digital currency that was introduced to the world in 2008. The underlying technology behind bitcoin as well as the other cryptocurrencies which followed it is blockchain technology. Nowadays, collaborative engineering is gaining tremendous popularity, especially in the information technology industry. In the future, technological giants may have to work with each other to complete large projects of great importance and significance. In such projects, trust management is crucial. With multiple parties involved in such projects, it is important that no party monopolizes the project and adds patches which are not known to the other parties. In addition, no party should have the authority to disclose the full contents of the work to a third party. In this paper, we propose a blockchain based solution to the above issues. The only authentic version of the project would be the one maintained in the blockchain. Collaborators would broadcast the changes they make, and it would be added to the blockchain only after consensus from all parties. We use a threshold key system on private keys so that consensus is achieved. Our proposed system guarantees secure release of the project as well as non-repudiation to collaborators.**

Keywords—Blockchain, Smart Contracts, Trust Management, Consensus, Smart Collaboration, Collaborative Development

I. INTRODUCTION

Blockchain[15], an innovation which has taken the virtual world by storm is a form of an electronic distributed ledger which allows multiple parties to transact between each other a plethora of data in various forms such as confidential information, cryptocurrencies, personal identities in an encrypted fashion which has proven nearly impossible to be tampered with.

Blockchain uses hash tables as data structures which are implemented on timestamp servers and each transaction on the timestamp service is verified using the Proof of Work [15][18]methodology. A timestamp server [16] works by taking a hash of a block of items to be time stamped and widely publishing the hash, such as in a newspaper or Usenet post. The timestamp proves that the data must have existed at the time in order to get into the hash. Each timestamp includes the previous timestamp in its hash, forming a chain, with each additional timestamp reinforcing the ones before it [16]. The blockchain structure is described in Fig.1. below [17].

This work was partially sponsored by Society for Computer Technology and Research (SCTR's) Pune Institute of Computer Technology, Pune, India.

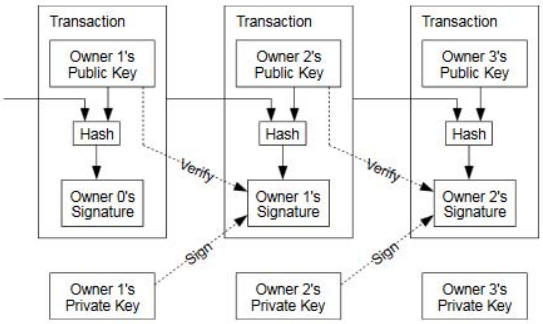

Fig.1. Blockchain Structure

The most relative and well understood use-case of the blockchain technology is bitcoin[16]. Through a clever combination of cryptography and game theory, the Bitcoin 'blockchain' – a distributed, public transaction ledger – could be used by any participant in the network to cheaply verify and settle transactions in the cryptocurrency.

Blockchain technology has been applied in the domains of healthcare, electronics, manufacturing, education, economics, social networking, etc. In our proposed model, we apply blockchain for smart collaboration in the software development process.

The rest of the paper is organized as follows. In Section II, we present a brief literature survey of the related work. Section III focuses on our proposed work, in which we describe the various elements of our system, the working of the system and the variation of the ECDSA signature scheme used. In Section IV, we analyse our model and state the improvements as well as the shortcomings as compared to the traditional models. We conclude in Section V.

II. RELATED WORKS

Since the introduction of blockchain and cryptocurrencies, a lot of research has been carried out in applications of blockchain in areas other than cryptocurrencies such as smart contracts, collaborations, social networking, etc.

978-1-5386-5850-5/18/$31.00 ©2018 IEEE
DOI 10.1109/CSCloud/EdgeCom.2018.00029

IEEE computer society

Smart contracts have been used extensively in the past few years. In [3], an automated smart contract management scheme has been described in which a hierarchy of common secrets are used to facilitate secure hierarchical communication.

In Hawk[2], a decentralized smart contract system for financial transactions is proposed, in which the transaction details are hidden from the public, but yet are a part of the blockchain. Hawk guarantees on-chain privacy as well as contractual security, with the contract programmer having no knowledge of cryptography.

Smart collaborations or partnerships are important tools for advancement of technology, especially in the fields of IT and research and development. Smart collaborations demonstrates in many research that people will learns and acquire from each other complementary technologies, skills, product, knowledge and technology updates [4][5]. Since 1998, after the IT boom, there has been a growing trend of inter-firm R&D collaborations [5].

Smart collaboration technologies have been developed in various domains. The authors in [7] have proposed a system for smart manufacturing. An integrated teaching learning smart collaboration system has been implemented in [4]. In [8], a tool for collaboration regarding event management operations has been implemented. In [14], the authors put forward an application of blockchain technology in the cooperative development of power electronic devices. he participants are to design the power supply topology in accordance with the parameters required by the producer. Many of the mechanisms like the one proposed in [9] work on real time chat services.

Smart collaboration has also been applied in software development. In [6], a technology is introduced in which, using collaborative aids, various collaborators can work together on complicated tasks and arrive at a consensus. The authors in [10] have designed an application for real time collaboration status monitoring as well as controlling. Another approach for web based collaboration for software engineering has been suggested in [11].

The mechanisms discussed above have several shortcomings. First, they are not secure from standard network attacks such as denial of service, man in the middle attacks, etc. Secondly, they all have a centralized authority. If the central authority is compromised, there is a risk to the project and the underlying intellectual property. Rogue collaborators and authorities may lead to unsecured and unlicensed release of data.

III. PROPOSED WORK

A. Background

A blockchain based approach can be implemented for smart collaboration between multiple entities. All collaborations in the project will be maintained in the form of a transaction in the blockchain. Due to the properties of the blockchain, no central entity can game the system and gain control of the project. In addition, it is not possible to revert any previous transactions after committing them to the blockchain. In other words, an entity cannot undo a particular update and later claim that they did not do the update.

B. System Elements

In our proposed model, there are N parties collaborating on a certain project. We distinguish between three different types of entities: collaborators, administrators and miners. Each party has several collaborators and one administrator. Collaborators are the entities working on the project. They are the ones who perform the commits on the code. Administrator is responsible for authorizing an action. An administrator is a representative of the party. This is still a decentralised system, as the authority is split equally to each administrator. Miners are the entities who verify transactions and maintain the blockchain. Miners help to verify the integrity and authenticity, as well as enforce smart contracts. A schematic diagram is shown below in Fig.2.

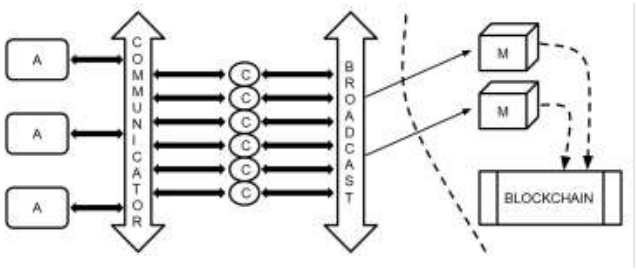

Fig.2. Proposed System Elements

C. System Working

We propose that each collaboration is encoded as a transaction message. In the proposed approach, we use a multiple signature scheme. Every administrator has their own private key which is used to digitally sign the transaction message.

The transaction message will be accompanied by the files which have been changed, as referenced in the updated files parameter in the message above. In addition, a hash of the updated files will be included so that tamper across the network does not take place. This message will be digitally signed by the collaborator.

Each transaction message is divided into chunks which have to be digitally signed by administrators. An administrator is responsible for signing one chunk. Unless all the administrators approve, the signature is not complete and the transaction is not valid.

The system working can be described in brief as follows:

1. Let a collaborator C_i contribute to the project under consideration. He/she will update all the changes and create a transaction message. This message will be digitally signed by C_i. The format of the transaction message is shown in Fig.3.

2. The message T will be divided into N equal parts $T_1, T_2 \ldots\ldots T_N$.

3. Then, C_i. will broadcast the transaction message T_i to the i^{th} administrator. Every administrator receives a part of the transaction message.
4. Along with the transaction message, a copy of the changed files is also sent. This is accompanied by the MAC for prevention of tampering.
5. After verifying the collaboration in the project, every administrator encrypts T_i with his private key and sends it back to the collaborator who initiated the request.
6. After receiving the signatures from all administrators, the transaction is broadcasted for inclusion in the blockchain.
7. To verify a transaction message, it must be divided into N parts again and check whether it was the administrators who signed it. This verification is carried out by the miners. The signature scheme is described in Fig.4.
8. When verified, the transaction is added to the blockchain. The updated files, along with the hash MAC agreed upon by all administrators is also included in it. Once a part of the blockchain, it can be considered to be an authentic part of the project.

This working is described in Fig.3 below.

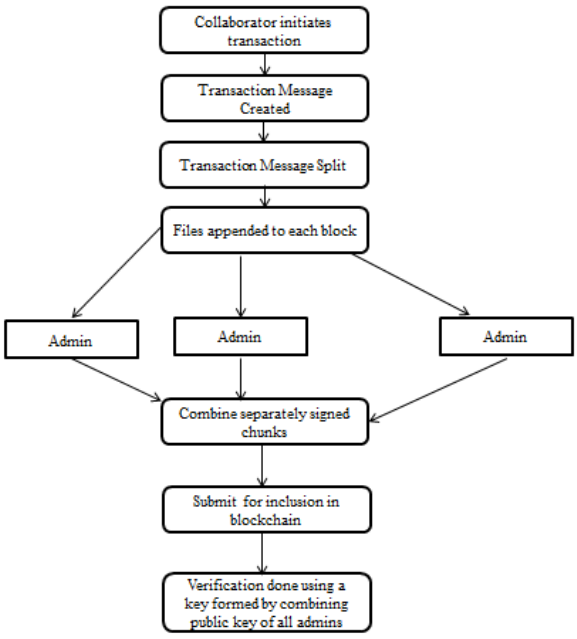

Fig.3. System Working

D. Proposed Message Formats

We propose the following format for the transaction message.

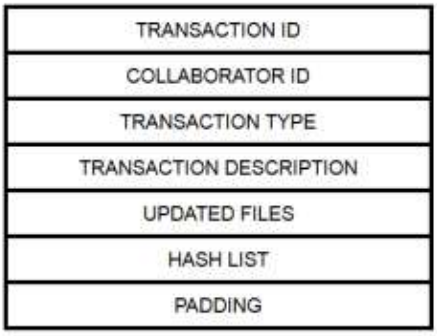

Fig.4. Transaction Message Format

The fields depicted in Fig.4. are described in brief here.

- ***Transaction ID:*** This is a unique identifier for a particular transaction. A transaction ID will be linked to the set of files modified. This value is unique system wide.
- ***Collaborator ID:*** The unique identifier to identify the collaborator who is proposing this transaction in the project.
- ***Transaction Type:*** Code to reflect the nature of the transaction such as insertion, deletion, modification, version update, release, etc.
- ***Transaction Description:*** A description of changes made in an informal way. This will function similar to a readme file.
- ***Updated Files:*** The list of files which have been modified as a result of this transaction.
- ***Hash List:*** The hash values computed after each file has been modified. The hash list contains an entry for each file modified.
- ***Padding:*** Data bits added at the end to make the message into a suitably sized packet.

E. Signature Scheme

Due to the use of multiple signatures, no party can modify any part of the work without all other parties knowing about it and approving it. In addition, this would also prevent non-consensual release of the project, since the project release message will have to be signed by all parties.

Every administrator has his own pair of public and private keys. Public keys are distributed by some common mechanism. As the signing is done chunk-wise, the signature verification is also carried out chunk-wise.

This can be considered to be a threshold key system[12], with the threshold as 1, which means that all parties must sign the transaction. We can also assign different weights to each administrator, and the signature considered valid only when a certain weight has been matched [13].

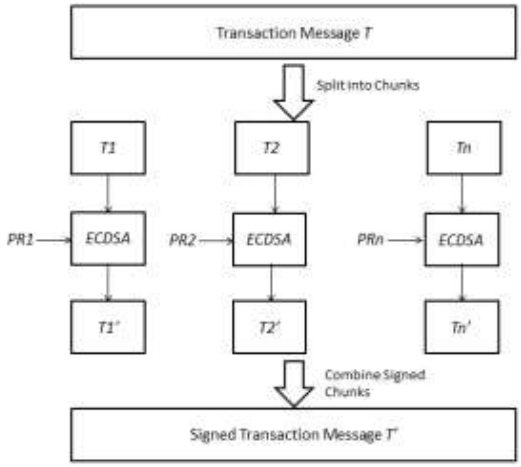

Fig.4. Signature Scheme

The proposed signature scheme is shown in Fig.5. above. We use ECDSA for digitally signing the message.

In Fig.4. above, *PR1, PR2, ...PRn* represent the private keys of the admins respectively. The figure depicts the signing mechanism. The verification mechanism will be carried out by splitting *T'* into chunks and verifying the signature on each chunk *Ti'* with the public key *PBi* of the i^{th} administrator.

IV. ANALYSIS

The proposed blockchain based system for smart collaboration has several advantages over conventional systems for the same. The model offers the following properties and features.

- ***Consensus**:* Any update in the project has to be approved by all of the administrators. Thus, no party can make a change of its own will without approval from the others. All changes are thus made in consensus.
- ***Authentication:*** As private keys are used to generate the digital signatures using ECDSA, identity of the collaborator as well as administrator can be verified.
- ***Secure Release:*** The project cannot be released until the release transaction message has been approved by all of the parties involved. This offers a great level of security to the underlying intellectual property involved.
- ***Authenticity:*** A particular update or patch in the software will be considered valid and secure only if a corresponding transaction exists in the blockchain. If not, the patch could be a rogue one with malicious intentions.
- ***Decentralization:*** This is the fundamental service provided by the blockchain. All miners will maintain a copy of the transaction ledger. Thus, there is no single point of compromise in the system which can lead to a single party gaining monopoly.
- ***Non Repudiation:*** A collaborator cannot claim that he did not make a particular update if the corresponding transaction exists in the blockchain. This is achieved using both digital signatures as well as blockchain consensus algorithms.

As with any system, the proposed model has some loopholes and vulnerabilities as well. Many of them are inherent characteristics of the blockchain, and hence are not discussed here. The drawbacks of the proposed approach are:

- ***Increased Latency**:* The transaction messages are first broadcast to administrators, and then to the miners for inclusion in the blockchain. This is a time consuming process, and reduces the speed of the system.
- ***Increased Traffic:*** A large number of messages are exchanged over the network. The messages are bulky as they carry the updated files as well. This increases network traffic and may lead to collision or packet loss.
- ***Complexity:*** The process involves generation, signing and verification of signatures using ECDSA several times. In addition, hash values have to be computed. Transaction messages have to be encrypted. These processes increase the computational complexity of our system.

We aim to solve these issues to improve performance in future works.

V. CONCLUSION

Blockchain, which is a distributed ledger of transactions has various applications in the fields of commerce, economics, healthcare and information technology. With the decentralization in blockchain, there is no more a dependency on centralized entities or trusted third parties.

In this paper, a novel application of the blockchain has been suggested. By encoding each update on a project in a collaborative environment in the blockchain, we can achieve security, authentication, non-repudiation, etc. This will lead to a smoother and more secure collaboration process.

ACKNOWLEDGEMENTS

The authors would like to express their gratitude to Dr. Prahlad Kulkarni, Principal, Pune Institute of Computer Technology, Pune for providing institutional support for this work.

The authors would also like to thank Dr. Rajesh Ingle, Head of Department, Department of Computer Engineering, Pune Institute of Computer Technology, Pune, India. This work would not have been possible without his support.

REFERENCES

[1] Xueping Liang, Juan Zhao, Sachin Shetty, Jihong Liu, Danyi Li, "Integrating blockchain for data sharing and collaboration in mobile healthcare applications", 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)

[2] Ahmed Kosba, Andrew Miller, Elaine Shi, Zikai Wen, Charalampos Papamanthou, “Hawk: The Blockchain Model of Cryptography and Privacy-Preserving Smart Contracts”, 2016 IEEE Symposium on Security and Privacy

[3] Craig Wright, Antoaneta Serguieva, “Sustainable blockchain-enabled services: Smart contracts”, 2017 IEEE International Conference on Big Data (BIGDATA)

[4] Ruhani Ab Rahman, Norhayati Ahmad, Murizah Kassim, Cik Ku Haroswati Cik Ku Yahaya, “Case study of a smart collaboration: FEE-UiTM & Cisco Network Academy experience”, 2009 International Conference on Engineering Education (lCEED 2009)

[5] J. Hagedoorn, "Inter-fum R&D partnerships: An overview of major trends and patterns since 1960", Research Policy, 31(4), pp. 477-492, 2002.

[6] S. Streng , R. Atterer, "Non-Invasive Collaboration Aids: Supporting Group Learning With Pervasive and Ambient Technologies, 2008 International Conference on Computer Science and Software Engineering, 2008.

[7] J.J Kim, S. W. Lee, W. P. Hong, E.G. Kang, D. H. Jun, "Data Application Plan in the Collaboration", International Conference on Smart Manufacturing Application April. 9-11, 2008 in KINTEX, Gyeonggi-do, Korea, ICSMA 2008.

[8] Santi Caballé, Fatos Xhafa, Jordi Raya, Leonard Barolli, Kazunori Uchida, “ Towards a Platform-Independent Event Management Model for Web Collaboration”, 2014 International Conference on Intelligent Networking and Collaborative Systems.

[9] Bogdan Ionescu, Cristian Gadea, Bogdan Solomon, Mircea Trifan, Dan Ionescu, Vasile Stoicu-Tivadar, “A chat-centric collaborative environment for web-based real-time collaboration”, 2015 IEEE 10th Jubilee International Symposium on Applied Computational Intelligence and Informatics.

[10] Lei Wu, H. Sahraoui, “Supporting Web Collaboration for Cooperative Software Development”, IEEE/WIC/ACM International Conference on Web Intelligence 2004

[11] J. Peng and K. H. Law, “A Prototype Software Framework for Internet-Enabled Collaborative Development of a Structural Analysis Program”, Engineering with Computers, Springer-Verlag, 2002, 18: pp. 38–49

[12] S. Goldfeder, R. Gennaro, H. Kalodner, J. Bonneau, J. A. Kroll,E. W. Felten, and A. Narayanan, “Securing bitcoin wallets via a new DSA-ECDSA threshold signature scheme,”, 2016 *Available at https://www.cs.princeton.edu/ steve-nag/thresholdsigs.pdf.*

[13] Pratyush Dikshit, Kunwar Singh, “Weighted threshold ECDSA for securing bitcoin wallet”, in 2017 ISEA Asia Security and Privacy (ISEASP), 2017

[14] Yinxin Yan, Bin Duan, Ying Zhong, Xiangshuai Qu, “Blockchain technology in the internet plus: The collaborative development of power electronic devices”, IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society

[15] Catalini C, Gans JS, “Some simple economics of the blockchain”, National Bureau of Economic Research; 2016 .

[16] Satoshi Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” *Available at http://bitcoin.org/bitcoin.pdf*

[17] Yi Liu, Ruilin Li, Xingtong Liu, Jian Wang, Chaojing Tang and Hongyan Kang, “Enhancing Anonymity of Bitcoin Based on Ring Signature Algorithm”, 2017 13th International Conference on Computational Intelligence and Security

[18] Ethereum Foundation. “Ethereum’s white paper,”, 2014, *Available at https://github.com/ethereum/wiki/wiki/White-Paper*

[19] Inderpal Singh Mehta, Arnav Chakraborty, Tanupriya Choudhury, Mukul Sharma, “Efficient Approach towards Bitcoin Security Algorithm”, 2017 International Conference on Infocom Technologies and Unmanned Systems (ICTUS'2017)