Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/6/20230851

Vulnerabilities and attacks on the blockchain software engineering landscape

Maheshwari V and Prasanna M

School of Information Technology and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India.

prasanna.m@vit.ac.in

Abstract. Blockchain is also known as Distributed Ledger Technology (DLT) and real transparencies of the history of digital assets by decentralization and encryption. It guarantees that the user's data never be erased, making it impossible to alter or falsify. Some people know that the "Blockchain revolution" can be compared with the internet and the web in their early days. As a result, all software development around blockchain is growing incredibly. Most software engineers are interested in blockchain technologies as they rush to develop unregulated software. Although some research has been performed on blockchain security and privacy concerns, a thorough analysis state of blockchain security is lacking. This article explores current problems and new principles for blockchain-oriented software engineering (BOSE) and discusses the need for new software engineering practices in the blockchain industry. Also, examine the solutions to improve blockchain protection, which might have been used to develop different blockchain applications, and suggest a few potential directions for moving research into this area.

Keywords: Blockchain-oriented software engineering (BOSE), Distributed Ledger Technology (DLT), Decentralization, Security

1. Introduction

Blockchain is a technology used to create a permanent, unchangeable record of transactions. It is a digital ledger of economic transactions that can be programmed to record financial transactions and virtually everything of value [1]. This technology has been around for about 10 years and has seen some major use in industries like banking and finance, but it's still not mainstream. Blockchain could revolutionize the way we do things and make everything more secure. The software system's design defines how its components are structured or connected. The blockchain is a part of a distributed software framework implementation layer. It aims to ensure that a specific non-functional feature of a distributed software system is to achieve and retains integrity. The integrity and trust in the peer-to-peer system are missing, accomplished using blockchain. Major integrity threats are technical failures and malicious peers. To make integrity, one should know the peers and the trustworthiness of the peers [2]. It is essential to consider what integrity measures and standards are applied at the start of any blockchain development project. When digital currencies represented real monetary value, hacks and assaults began. The websites enable the storage and exchange of digital currencies for other, legal or illegal, significant targeted attacks. The biggest was the MtGox attack in early 2014, which resulted in

DOI: 10.54254/2755-2721/6/20230851

a $600 million loss and the Bitfinex seizure in August 2016, losing $65 million [3]. Another notable achievement was the DAO's continued funding in June 2016, contributing to removing ether's digital currency assets by $50-60 million.

The Ethereum community saved itself from this attack by executing a hardware Ethereum fork, which recovered the robbed ethers and returned them to the original owners. These attacks are due to shoddy software development practices [4].In this regard; we identified the significant challenges in state-of-the-art blockchain-based software engineering. We define the major BOSE problems and their issues. We also provide experts from SWEBOK3 to identify the issues involved correctly [5]. Defining new professional roles, demanding testing activities and new software architecture tools are today's challenges in software engineering. We are taking a step forward by proposing new directives based on an inventoried corpus of blockchain-oriented software repositories, of which Moody's blockchain report data has been detected for 2016 [6]. Figure 1 shows the Blockchain-oriented software engineering challenges and the complicated issues.

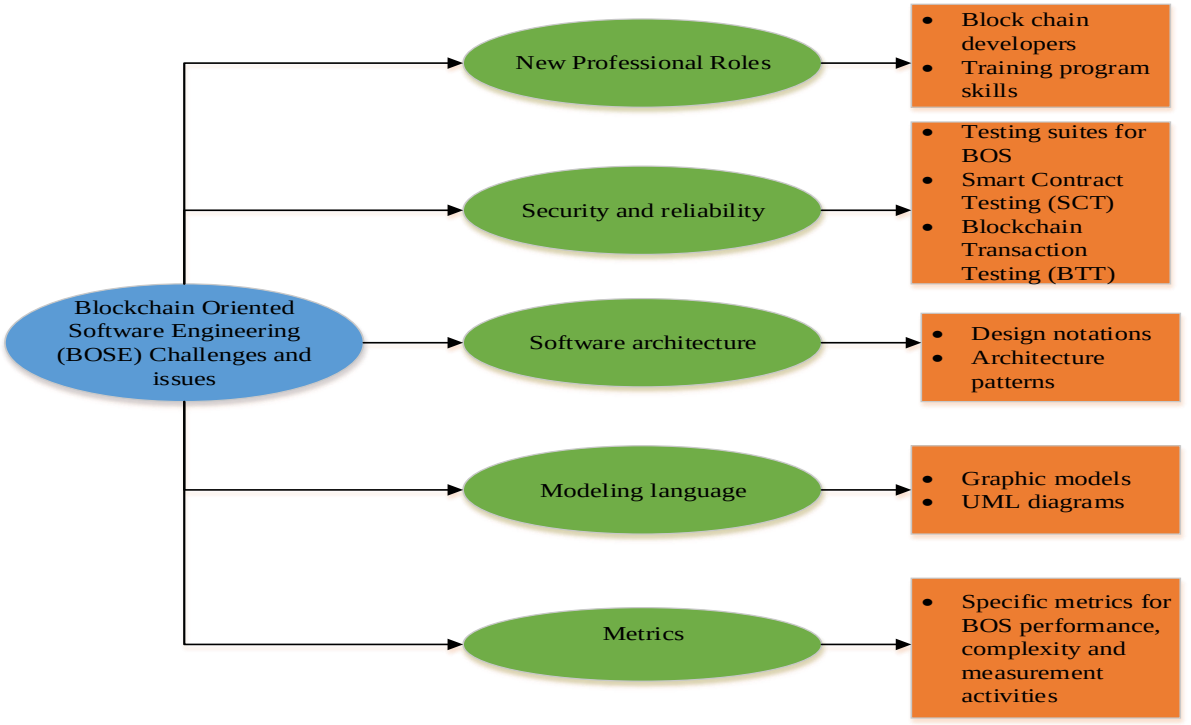

Figure 1. Blockchain-oriented software engineering (bose) challenges and issues.

The high efficiency, security standard, and scalability requirements for a distributed system can only be met with careful system design and thorough testing, using standard software engineering tools and techniques and novel approaches to testing that take into account the unique characteristics of blockchain-based distributed ledgers. Destefanis et al. recommend implementing and concentrating on Blockchain-oriented software engineering (BOSE). Blockchain-based technology provides a desirable source of attackers with high and low complexity. Low architecture and design possibilities and incomplete development tools indicate that safety-conscious developers are prone to potential safety vulnerability. The immutable aspects of blockchain technology prohibit the retrieval or effectively stop it from evolving after deployment.

The development team is urged to develop a forward-looking strategy to recommend practice software engineering, secure development, and substantial bug detection check once we enter the system. Researchers have demonstrated that the controversial parity attack resulting in the freezing of

DOI: 10.54254/2755-2721/6/20230851

162 million USD can be alleviated by consistently applying good practices in software engineering. The collection of requirements and the task selection among team participants are other essential software engineering elements that significantly impact blockchain development [7]. When the industry is hugely competitive, and new products take centre stage nearly every month, designing new attractive features is a significant challenge. The justification for the excessive sophistication of the project allocation in Blockchain native applications (as related to standard software development) is the fundamental nature of the development team, which is dynamic and loosely controlled because most projects are open source projects. We need to understand current software engineering activities or the lack of blockchain service (BCS) projects to ensure efficient blockchain applications. The developers should follow fair practices consistently. We have no proper study of methodology for software engineering and blockchain development projects to the best of our ability that would expose the truth to the research community's attention [8]. To provide the latest missing definition of blockchain-centred projects, we have set a goal for carrying out the first systematic study to analyze the software engineering process: requirement analysis, task assignment, testing, and verification [9].

The anonymous user took two measures to exploit the vulnerability. First, the intruder must own the smart contract library (as it was created and uninitialized) and only call the initialization function. The suicide feature was added later, destroying the library and stopping smart agreements with the library from working as all delegates' calls were finished in the dead smart contract system [10].

This case demonstrated the need for blockchain-driven software development to avoid or mitigate these scenarios. BOSE'S goal is to pave the way for smart contract applications to be managed, testable, and verifiable.

2. Blockchain cybersecurity vulnerabilities

Many of today's online company transfers are performed using blockchain technologies. The author builds on prior analyses of security flaws of 60 actual attacks and frauds in the blockchain. Blockchain systems investigate five categories of attacks: non-based attacks, authority attacks, asset-based attacks, insider attacks, asset-based attacks, third and fourth-party attacks, and quantum attacks. As a result, the outcome motivated the researchers to study the countermeasures needed to neutralize these threats [11]. The Blockchain cybersecurity vulnerabilities and their attacks are explained in table 1. Attacks like double-spending, transaction malleability, Sybil attacks, attacks on BC networks, mining pools, and bugs in Smart contracts are the most widespread attacks in the application [12].

Double spending is an attack in which the attacker spends coins on more than one recipient. This attack can be made by reversing or creating a fraudulent transaction. To prevent the double-spending attack, the blockchain uses cryptographic algorithms to verify transactions and ensure that there are not any duplicate transactions. Transaction malleaility is a bug in the bitcoin protocol that allows for transaction bitcoins to be modified after they have been sent. It is not a problem with bitcoin itself but rather with how nodes on the network process transactions [13].

Table 1. Blockchain cybersecurity vulnerabilities and attacks.

Cybersecurity Vulnerabilities	**Attacks**
Client Vulnerabilities	Digital Signature Vulnerability, Hash function Vulnerability, Mining malware Vulnerability, Addresses Vulnerability, and Software Flaws.
Consensus Vulnerabilities	51% Vulnerability, Alternative history attack, Finney attack, and so forth.
Mining Pool Vulnerabilities	Block withholding attack (BWH), Bribery attack, and so forth.
Network Vulnerabilities	Transaction Malleability attack
Smart Contract Vulnerabilities	Ethereum Virtual Machine (EVM) byte code Vulnerability, Solidity Vulnerability

DOI: 10.54254/2755-2721/6/20230851

A Sybil attack is a type of attack on a peer-to-peer network in which one or more malicious nodes create multiple identities, or "Sybils," to gain control of the network. The blockchain is a distributed ledger that stores data across its network and can be used to record transactions and track assets [14].

2.1. Blockchain security and development tools

Blockchain security is a comprehensive risk assessment procedure for a blockchain solution or network to ensure security [15]. Blockchain security is achieved via cybersecurity frameworks, security testing methodologies, and secure coding practices to protect a blockchain solution from online fraud, breaches, and other cyberattacks. This section describes blockchain security and development tools and its method in table 2 and table 3.

Table 2. Blockchain security tools.

Security Tool	**Method**	**Bytecode analysis**	**CLI**	**WUI**	**Solidity analysis**
Oyente	Symbolic execution	√	√	√	√
Securify	Formal verification	√	×	√	√
Gasper	Symbolic execution	√	**N/A**	**N/A**	×
Simple Analysis	Heuristics	√	×	√	√
Mythril	Concolic testing (Symbolic execution)	√	√	×	√
Smart check	N/A	√	×	×	√
Remix	Formal Verification		√	√	√
F*Framework	Formal Verification	√	√	×	√

2.2. Open research problems

No consistent tools can resolve BCS features in terms of functional, scalability, and security testing. Testing is an important part of development, and it's necessary to test blockchain applications for bugs and security. Regression testing is verifying that the latest changes to a software or system have not broken anything previously working.

Regression testing blockchain applications can be made in many ways, such as manual, automated, functional, integration, and stress tests. Particular consideration is given to the low response to integration or regression testing, provided that distributed development exists in many BCS Projects. Besides, there was a lack of knowledge of developers' advanced and automated testing methods like fault tolerance in the application, property-based testing, data transparency and resource sharing in supply-chain, smart contract testing, mutation testing, automated testing for DApps, Stress testing, performance testing, automated security testing, application fuzz testing.

The lack of testing tools, particularly for integration, and regression testing, which is the primary trigger for the diversion from the prescribed practice, requires the creation of SE tools and researchers to be taken into account. Scientists and researchers also encounter the demands of blockchain guidelines for professional software engineering.

DOI: 10.54254/2755-2721/6/20230851

Table 3. Blockchain development tools.

Tool	Language support	Use
Solidity	C++, Python, JavaScript	Create Contracts for voting, blind auction, multi-signature wallet, crowdfunding
Geth	Go Programming Language	Transferring tokens, mining ether tokens, creating a smart contract, exploring block history
Mist		Crete Ethereum, store ether token, execute smart contract
Solidity Compiler	C++, JavaScript	Convert solidity script to a readable format for EVM
Remix	Browser-based BC tool	creation and deployment of SC
Metamask	Browser extension	Serve ether, and ERC-20 assets interact with Ethereum Dapps
Truffle	Framework	Custom deployment of new SC, development complex Ethereum dapps, automated CT using chai and mocha
Ganache	Corda distributed application development	Create your private Ethereum BC to test dapps, command line, advanced mining control, and built-in block explorer

Recently, Bitcoin has gained much favor and can be called one of the new "big data" subjects. The unyielding impact it has on businesses and individuals is an undisputed fact, but they disagree about scalability, security, and sustainability. Accordingly, granting blockchain technology functionality to provide more secure and expedient services is necessary. Still, it must also consider the protection and privacy aspects and make allowances for several blockchain implementations, from banking, healthcare, Internet of Things (IoT) to the general public and society. Some experiments have concentrated on using the blockchain structure in different areas.

However, an in-depth study on these topics has not been conducted till now. This article intends to undertake a deep and extensive analysis of different BC tools, algorithms, and the possibilities and problems concerning protection and privacy in the blockchain. Moreover, we explore how the blockchain will develop in the future.

3. Conclusion

This article has shown a new direction for successful software development looked for with blockchain-oriented software engineering (BOSE). New professional positions, improved security and complexity, modelling and verification frameworks, and specific metrics are required to push blockchain applications to a higher level of complexity. Before embarking on the blockchain application in the business, managers should know their security flaws and weak points. Organizations must clearly understand block chain’s security risks before it is considered for enterprise adoption. The goal of BOSE is to create a bridge between conventional software engineering and blockchain application, to develop new Adhoc methodologies, fault analysis, pattern quality metrics, and security. In our view, Software engineering combined with blockchain helps create safer and more reliable applications, which appears to be very effective in developing future BOSE Security.

DOI: 10.54254/2755-2721/6/20230851

References

[1] Sathishkumar, V. E., Park, J., & Cho, Y. (2020). Using data mining techniques for bike sharing demand prediction in metropolitan city. *Computer Communications*, *153*, 353-366.

[2] Senthilkumar, K., & Easwaramoorthy, S. (2017, November). A Survey on Cyber Security awareness among college students in Tamil Nadu. In *IOP Conference Series: Materials Science and Engineering* (Vol. 263, No. 4, p. 042043). IOP Publishing.

[3] VE, S., & Cho, Y. (2020). A rule-based model for Seoul Bike sharing demand prediction using weather data. *European Journal of Remote Sensing*, *53*(sup1), 166-183.

[4] Krishnamoorthy, N., Prasad, L. N., Kumar, C. P., Subedi, B., Abraha, H. B., & Sathishkumar, V. E. (2021). Rice leaf diseases prediction using deep neural networks with transfer learning. *Environmental Research*, *198*, 111275.

[5] VE, S., Shin, C., & Cho, Y. (2021). Efficient energy consumption prediction model for a data analytic-enabled industry building in a smart city. *Building Research & Information*, *49*(1), 127-143.

[6] VE, S., Park, J., & Cho, Y. (2020). Seoul bike trip duration prediction using data mining techniques. *IET Intelligent Transport Systems*, *14*(11), 1465-1474.

[7] Easwaramoorthy, S., Sophia, F., & Prathik, A. (2016, February). Biometric Authentication using finger nails. In *2016 international conference on emerging trends in engineering, technology and science (ICETETS)* (pp. 1-6). IEEE.

[8] Easwaramoorthy, S., Thamburasa, S., Samy, G., Bhushan, S. B., & Aravind, K. (2016, April). Digital forensic evidence collection of cloud storage data for investigation. In *2016 International Conference on Recent Trends in Information Technology (ICRTIT)* (pp. 1-6). IEEE.

[9] VE, S., & Cho, Y. (2020). Season wise bike sharing demand analysis using random forest algorithm. *Computational Intelligence*.

[10] Subedi, B., Sathishkumar, V. E., Maheshwari, V., Kumar, M. S., Jayagopal, P., & Allayear, S. M. (2022). Feature learning-based generative adversarial network data augmentation for class-based few-shot learning. *Mathematical Problems in Engineering*, *2022*.

[11] Rajalaxmi, R. R., Narasimha Prasad, L. V., Janakiramaiah, B., Pavankumar, C. S., Neelima, N., & Sathishkumar, V. E. (2022). Optimizing Hyperparameters and Performance Analysis of LSTM Model in Detecting Fake News on Social media. *Transactions on Asian and Low-Resource Language Information Processing*.

[12] Shanthi, N., VE, S., Upendra Babu, K., Karthikeyan, P., Rajendran, S., & Allayear, S. M. (2022). Analysis on the Bus Arrival Time Prediction Model for Human-Centric Services Using Data Mining Techniques. *Computational Intelligence & Neuroscience*.

[13] Pavithra, E., Janakiramaiah, B., Narasimha Prasad, L. V., Deepa, D., Jayapandian, N., & Sathishkumar, V. E. (2022). Visiting Indian Hospitals Before, During and After COVID. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*.

[14] Karrothu, A., Anilkumar, C., & Sathishkumar, V. E. (2022). An Escrow-Free and Authenticated Group Key Management in Internet of Things. In *Smart Intelligent Computing and Applications, Volume 2* (pp. 505-512). Springer, Singapore.

[15] Chen, J., Shi, W., Wang, X., Pandian, S., & Sathishkumar, V. E. (2021). Workforce optimisation for improving customer experience in urban transportation using heuristic mathematical model. *International Journal of Shipping and Transport Logistics*, *13*(5), 538-553.

[16] Zhang, M., Wang, X., Sathishkumar, V. E., & Sivakumar, V. (2021). Machine learning techniques based on security management in smart cities using robots. *Work*, *68*(3), 891-902.