International Research Journal of Education and Technology
Peer Reviewed Journal
ISSN 2581-7795

Blockchain Technology and Its Security

Thiruvankatasamy S[1], Padma P[2]

[1]Assistant Professor, [2]PG Scholar, Department of Computer Science and Engineering,
Nandha College of Technology, Perundurai 638 052, Tamilnadu, India

Abstract

Blockchain technology is an emerging technology. Over recent years, blockchain technology has been attracting much attention from both scholars and practitioners in the supply chain management (SCM) domain. The idea of Blockchain technology has been originally developed to create cryptocurrencies (Nakamoto, 2008) who helped develop the first bitcoin software and introduced the concept of cryptocurrency to the world. Blockchain technology is a digital ledger that stores records of a transaction in a decentralized manner, usually with no central authority. It is organized in an append-only, sequential chain of blocks using cryptographic links and is designed to be tamper-resistant and tamper-evident. The technical characteristics of Blockchain technology enhance the security of transactions while allowing to create a system of shared trust and eliminating the problem of single point of failure.

Keywords: Blockchain, security perception, consensus algorithm, smart contract, risk.

1. Introduction

A primary objective of blockchain technology is to address information security and efficiency issues related to existing information sharing systems. Blockchain technology is a distributed database where all assets (tangible or intangible) are digitally encoded. This digital encoding helps easy registering, tracking, and trading through private keys provided on the blockchain. At the same time, Blockchain technology is facing an increasing number of cyberattack challenges. This technology can help organizations manage and distribute digital data by using mutually distributed ledgers. Literature shows that blockchain technology has four key components. These components include non-localization (decentralization), security, auditability, and smart execution [4]. This technology initially focuses on sharing and executing digital events among given blockchain. Furthermore, there are many advantages of using Blockchain technology. However, it still has many associated risks [2]. One of the major advantages of using Blockchain technology is a decentralized system. A decentralized system works without involving any third party or core administrator. Also, any data entered in the Blockchain technology system cannot be altered or deleted which helps in ensuring transparency and immutability. Blockchain technology system processing is much faster as compared to traditional systems. Blockchain technology system reduces processing time from 3 days to approximately several minutes or even seconds.

However, despite these advantages, Blockchain technology has many associated risks and disadvantages. Blockchain technology systems consume high energy as a substantial amount of computer power is required to keep a real-time ledger and ensure transparency. Also, Blockchain technology systems have a significant amount of initial capital costs. Most importantly, the Blockchain technology system has a high risk of external cybersecurity threats including 51% attacks, double-spending attacks, and Sybil's attacks [11].

2. Blockchain Technology

2.1 Consensus Algorithm

As one of the desired blockchain features, anonymity also poses a problem when it comes to trust. How can it be 100% ensured that anonymous users are honest when they add transactions to a ledger? The answer is to validate every transaction to be legal (not malicious, double spending, etc.) and then put the transactions into a block. The agreement of adding a block to the blockchain is through consensus algorithms. These consensus algorithms take advantage of the fact that the majority of users on a blockchain have a common interest in keeping the blockchain honest. A blockchain system uses a consensus algorithm to build its trust and properly stores the transactions on the blocks.

A consensus protocol is essentially a set of rules to be followed by every participant. As a distributed technology without a universal trust, blockchain needs a distributed consensus mechanism for all participants to agree on the blockchain's current state. The blockchain's consensus is based on scarcity that controlling more of a scarce resource gives more control over the blockchain's operation. A number of unique consensus mechanisms have been designed for blockchains, which include Proof of Work (PoW), Prof of State (PoS), Delegated Proof of State (DPoS), Proof of Elapsed Time (PoET), Practical Byzantine Fault Tolerance (PBFT), Directed Acyclic Graph (DAG, Proof of Authority (PoA), Tendermint, Ripple, Scalable Byzantine Consensus Protocol (SCP), Proof of Bandwidth (PoB), Proof-of-Importance (PoI), Proof of Burn, Proof of Capacity, depending on their unique requirements.

PoW, PoS, DPoS, and PBFT are the most common consensus algorithms. DAG is the most different from other consensus algorithms. PoET is developed by Intel Corporation and used in Hyperledger Sawtooth. Thus, these six consensus algorithms are further described below.

Proof of Work (PoW) - PoW selects a problem that can only be solved by guessing. For example, when it is time to create and validate a full block, the problem is to guess a nonce value such that when using the transaction data and the nonce value as inputs for a hash function, its hash output needs to match the difficulty, e.g., beginning with four leading zeros. Every node (also called mining node) on the network is now guessing different nonce values randomly until one node first happens to find the nonce value that matches the difficulty.

Proof of Stake (PoS) – PoS is the second most prominent consensus method and requires fewer computations for mining than PoW. PoS solves time and electricity consumption problems that PoW has because the electricity requirement is associated with miners finding a nonce and this process needs to take some time.

Delegated Proof-of-Stake (DPoS) -In DPoS, all token holders can vote for a number of delegates and can also delegate to other users with their voting power. The more tokens that the token holder has, the more voting power the token holder has. Then the delegates are responsible for validating transactions and blocks to secure the network.

Proof of Elapsed Time (PoET) - Intel Corporation developed PoET to enable a different way to determine a winner to mine a block. In PoET, each potential validation node requests a random waiting time which is generated on a trusted computing platform, e.g., Intel's SGX.

Practical Byzantine Fault Tolerance (PBFT) - Byzantine Fault Tolerance (BFT) is to solve a famous general problem that some generals are dishonest but needs to reach a correct consensus. PBFT is a consensus algorithm that optimizes BFT.

Directed Acyclic Graph (DAG) - DAGs are made up of vertices and the edges (the lines connecting them), which is different from other.

2.2. Smart Contract

The smart contract makes another beautiful part of blockchain that blockchain not only provides a distributed, unchangeable record of all the different events that have occurred, but also allows to write very non-subjective computer code that defines exactly how that process is going to be managed and what steps are going to be taken when that event occurs. One goal of the smart contract proposed in Ethereum was to break the limitations of Bitcoin.

Smart Contract also known as chain code:

- Program rules and decision points into blockchain transactions and processes.
- Automate transactions and ensure they are all following the same rules.
- Run on the blockchain.

2.3. Cryptography for Blockchain

Blockchain creates a layer of trust between untrusted parties to enable secure and trusted records and transactions to occur. Without blockchain to create trusted records and transactions, a third-party intermediary is necessary. blockchain uses cryptography and collaboration to create that trust and as a result, it eliminates the need for a centralized institution to act as an intermediary. Information on the blockchain is stored on the ledger using cryptography.

Blockchain makes use of some cryptography building blocks as below:

- Public Key Cryptography: Be used for digital signatures and encryption.
- Zero-Knowledge Proof: Demonstrate the knowledge of a secret without revealing it.
- Hash Functions: One-way pseudo-random mathematical functions. Merkle trees adopted the hash function to form one component of the block header.

Public key cryptography - It is used to prove that a transaction was created by the right person. In blockchain, the private key is kept in a digital wallet, either a hardware wallet (a physical device to store the private key) or any software wallet (e.g., a desktop wallet app, mobile wallet app, or web-wallet). A user accesses its private key to sign a message called a digital signature that will be transmitted to the blockchain, and its public key is to confirm that the message actually did come from the user. For example, in Fig. 1, the user hashes its transaction data into hash value 1 and then signs on the hash value 1 with its private key to generate the digital signature.

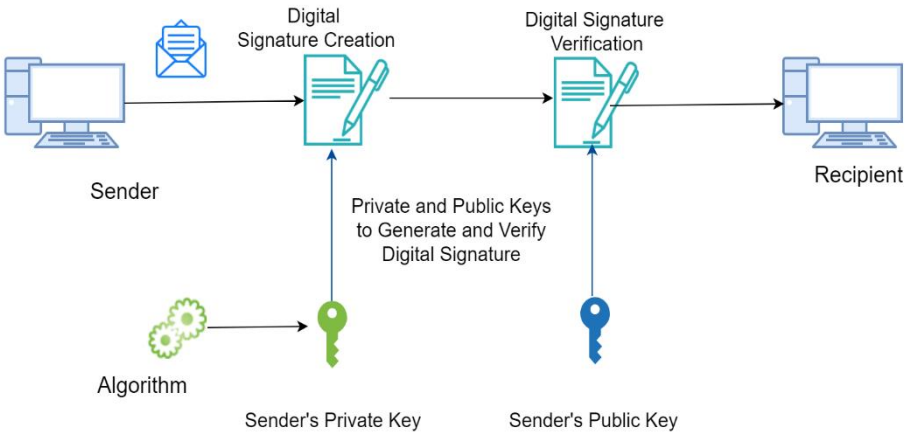

Figure 1. Creation and verification of a digital signature

Zero-Knowledge Proofs - One of the primary use cases for Zero Knowledge Proofs in blockchain is shown in the following. When a user makes a request to send another user some money, the blockchain naturally wants to make sure, before it commits this transaction, that the user who is sending money has enough money to send. However, the blockchain does not really need to know or care who is spending the money, or how much total money he/she has. In this case, the blockchain has zero knowledge about who the user is sending the money to and how much money the user has.

Hash Functions - Hash functions are a key technology used in the blockchain. A hash function is a mathematical equation with five important properties for cryptography:

- **Fixed size** - Hash functions can take anything as an input and create an output with a fixed size. This makes it possible to condense anything into a piece of data of a fixed size. So blockchains use hash functions to condense messages for digital signatures.

- **Preimage resistance** - Given an input, it is not hard to calculate a hash output. However, given the hash output, it is mathematically impossible to reverse-engineer the original input. In fact, the only possible way is to randomly input the data into the hash function until the same output is produced.
- **2nd preimage resistance** - If an input and its hash output are given, getting the second input that produces the same hash output is computationally infeasible.
- **Collision resistance** - Finding any two distinct inputs is computationally infeasible to produce the same hash output.
- **Big change** - If any single bit of the input is changed, it will produce an entirely different hash output.

3. Blockchain Applications

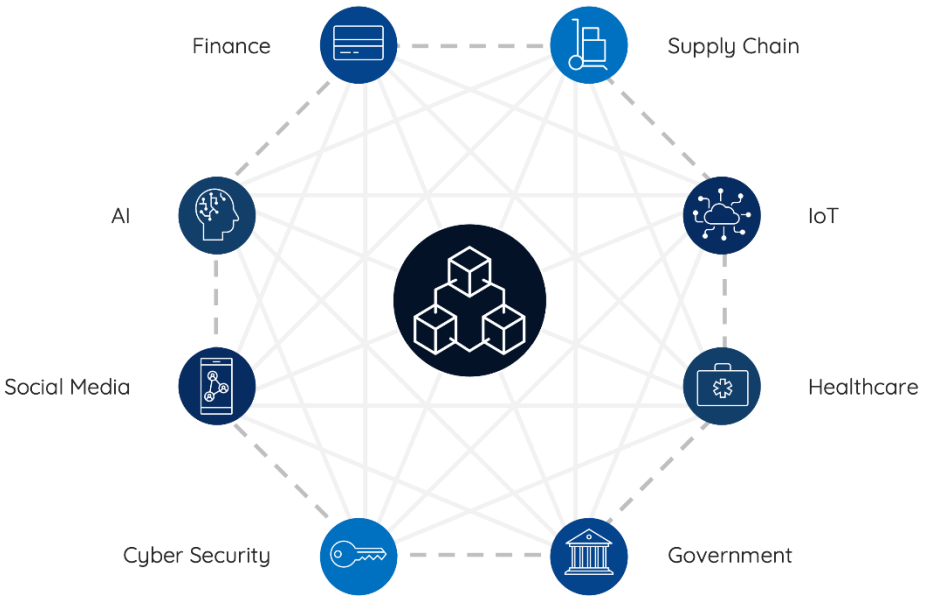

Figure 2. Applications of Blockchain

4. Security Risks and Attacks with Blockchain

As blockchain is decentralized without engaging any third party and needs to ensure trust in the trustless infrastructure, security on blockchain itself is worthy to conduct the research. This section will focus on security risks on blockchain technology, and a survey of real attacks and bugs on blockchain systems.

4.1. Real Attacks and Bugs on Blockchain Systems

In this paper, we survey some real attacks and bugs on blockchain systems to raise awareness of the need for security on blockchain systems. Users use exchange platforms to make transactions on blockchain, and on blockchain a private key is kept in a digital wallet. Hence, exchange platforms and wallets are parts of blockchain systems.

5. Security Measures for Blockchain

There are some security measures for blockchain. Some of them are described below.

5.1. Secure Smart Contract

In 2016, Luu et al. presented methods to enhance Ethereum operational semantics to reduce the smart contracts vulnerabilities [14]. In2016, Town Crier was developed to ensure only authenticated data be input into the smart contracts [15].

5.2. Detecting Malicious Codes and Bugs

In 2018, Jiang et al. proposed ContractFuzzer to fuzz smart contracts to detect vulnerability [17], Liu et al. presented ReGuard of a fuzzing-based analyzer in their demo paper to automatically detect the reentrancy bugs of the most common bug type in the smart contracts [18], and Hydra was developed by Breidenbach et al. to use bug bounties to enable rewarding of critical bugs and runtime detection [19].

6. Conclusions

This paper has first conducted a deeper survey on blockchain technology in terms of consensus algorithms, smart contracts, and cryptography for blockchain. Public key cryptography, Zero-Knowledge Proof, and hash functions used in blockchain have been described in detail for integrity, authentication, nonrepudiation, and payment address required in blockchain systems. This paper has then listed the comprehensive applications of blockchain. Further, the security of blockchain itself is a focus in this paper. It has surveyed many real attacks and bugs on blockchain systems and listed out their root causes. The paper has then presented the security measures in the areas of secure smart contract, detecting malicious codes & bugs. The users who use blockchain to do the transactions will pay more attention to the security of blockchain itself.

References

1. Abdelwahab, N. Ramadan, and H. Hefny, “Cybersecurity risks of blockchain technology,” *International Journal of Computer Applications*, vol. 177, no. 42, pp. 8–14, 2020.

2. H. Lu, K. Huang, M. Azimi, and L. Guo, “Blockchain technology in the oil and gas Industry: a review of applications, opportunities, challenges, and risks,” *IEEE Access*, vol. 7, pp. 41426–41444, 2019.

3. S. A. Abeyratne and R. P. Monfared, “Blockchain ready manufacturing supply chain using distributed ledger,” *International Journal of Research in Engineering and Technology*, vol. 5, no. 9, pp. 1–10, 2016.

4. S. Saberi, M. Kouhizadeh, J. Sarkis, and L. Shen, "Blockchain technology and its relationships to sustainable supply chain management," *International Journal of Production Research*, vol. 57, no. 7, pp. 2117–2135, 2019.
5. S. Gomathi, M. Soni, G. Dhiman, R. Govindaraj, and P. Kumar, "A survey on applications and security issues of blockchain technology in business sectors," *Materials Today: Proceedings*, 2021.
6. H. Hasanova, U. J. Baek, M. G. Shin, K. Cho, and M. S. Kim, "A survey on blockchain cybersecurity vulnerabilities and possible countermeasures," *International Journal of Network Management*, vol. 29, no. 2, p. 36, 2019.
7. A. A. Abd El-Latif, B. Abd-El-Atty, I. Mehmood, K. Muhammad, S. E. Venegas-Andraca, and J. Peng, "Quantum-inspired blockchain-based cybersecurity: securing smart edge utilities in IoT-based smart cities," *Information Processing & Management*, vol. 58, no. 4, article 102549, 2021.
8. S. Alonso, J. Basañez, M. Lopez-Coronado, and I. De la Torre Díez, "Proposing new blockchain challenges in eHealth," *Journal of Medical Systems*, vol. 43, no. 3, p. 64, 2019.
9. M. Andoni, V. Robu, D. Flynn et al., "Blockchain technology in the energy sector: a systematic review of challenges and opportunities," *Renewable and Sustainable Energy Reviews*, vol. 100, pp. 143–174, 2019.
10. S. J. Andriole, "Blockchain, cryptocurrency, and cybersecurity," *IT Professional*, vol. 22, no. 1, pp. 13–16, 2020.
11. J. Golosova and A. Romanovs, "The advantages and disadvantages of the blockchain technology," in *Paper presented at the 2018 IEEE 6th workshop on advances in information, electronic and electrical engineering (AIEEE)*, Vilnius, Lithuania, 2018.
12. M. K. Hasan, A. Alkhalifah, S. Islam et al., "Blockchain technology on smart grid, energy trading, and big data: security issues, challenges, and recommendations," *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 9065768, 26 pages, 2022.
13. Y. Himeur, A. Sayed, A. Alsalemi et al., "Blockchain-based recommender systems: applications, challenges and future opportunities," *Computer Science Review*, vol. 43, article 100439, 2022.
14. L. Luu, D.-H. Chu, H. Olickel, et al., Making smart contracts smarter, in: The 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS'16); 24–28 Oct 2016; Vienna, Austria, ACM, New York, NY, USA, 2016.
15. F. Zhang, E. Cecchetti, K. Croman, A. Juels, E. Shi, Town crier: an authenticated data feed for smart contracts, in: 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16); 24–28 Oct 2016; Vienna, Austria, ACM, New York, NY, USA, 2016, pp. 270–282.
16. B. K. Mohanta, D. Jena, S. S. Panda, and S. Sobhanayak, "Blockchain technology: a survey on applications and security privacy challenges," *Internet of Things*, vol. 8, article 100107, 2019.

17. B. Jiang, Y. Liu, W. Chan, ContractFuzzer: fuzzing smart contracts for vulnerability detection, in: Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering; 3–7 Sep 2018; Montpellier, France, IEEE, Piscataway, NJ, USA, 2018, pp. 259–269.
18. C. Liu, H. Liu, Z. Cao, et al., ReGuard: finding reentrancy bugs in smart contracts, in: The 40th International Conference on Software Engineering: Companion; 27 May–3 Jun 2018; Gothenburg, Sweden, IEEE, Piscataway, NJ, USA, 2018, pp. 65–68.
19. L. Breidenbach, P. Daian, F. Tramer, A. Juels, Enter the Hydra: towards principled bug bounties and exploit-resistant smart contracts, in: 27th USENIX Security Symposium; 15–17 Aug 2018; Baltimore, MD, USA, USENIX Association, Berkeley, CA, USA, 2018, pp. 1335–1352.
20. A. Vacca, A. Di Sorbo, C. A. Visaggio, and G. Canfora, "A systematic literature review of blockchain and smart contract development: techniques, tools, and open challenges," *Journal of Systems and Software*, vol. 174, article 110891, 2021.