Securing Blockchain Technology: A Comprehensive Analysis of Vulnerabilities and Mitigation Strategies

Bharathi Putta
Texas A&M University - Corpus Christi
Corpus Christi, Texas, USA
bputta@islander.tamucc.edu

Dulal Chandra Kar*
Texas A&M University-Corpus Christi
Corpus Christi, Texas, USA
dulal.kar@tamucc.edu

***Abstract*—This paper provides a comprehensive analysis of cybersecurity vulnerabilities in blockchain technology and presents mitigation strategies to counter them. The analysis covers recent attacks and their impact on blockchain security. The paper identifies the major types of blockchain vulnerabilities, including consensus algorithm vulnerabilities and wallet vulnerabilities for cryptocurrency applications. Mitigation strategies for these vulnerabilities are discussed, such as the use of consensus algorithms with improved security, the implementation of secure coding practices, and the adoption of multi-factor authentication for wallet access. The paper also highlights the importance of community efforts and collaboration between industry players in enhancing blockchain security. The analysis and strategies presented in this paper provide valuable insights for blockchain developers, researchers, and stakeholders seeking to secure their blockchain applications against cyber threats.**

***Index Terms*—Blockchain, Bitcoin, Consensus Algorithm, Smart Contract, Bitcoin, Cybersecurity.**

I. Introduction

Blockchain has emerged as a decentralized ledger system, which is becoming popular among business organizations of all different sizes, including those in the banking and financial services, healthcare, and real estate industries. In a traditional setting, a trusted third party is necessary to validate financial transactions. However, blockchain offers an alternative where a collective verification system replaces the intermediary, ensuring that all transactions are visible and immutable. This digital ledger incorporates a decentralized network where identical data copies can be maintained in multiple locations in real-time, accessible from anywhere. Once added to the ledger, a data block becomes tamper-resistant, as it is maintained across multiple nodes on the network and linked together with other data blocks [3]. Transactions by participants in the blockchain network are only recorded once in the distributed ledger, thus eliminating the duplication of effort often seen in traditional business networks.

By eliminating the need for third-party entities in transaction control, blockchain provides enhanced security, anonymity, and data integrity. Nevertheless, blockchain technology is not invulnerable to cyber-attacks [1]. Implementing risk management policies, threat analysis, and mitigation measures can help reduce the impact of an attack.

In the following, we cover the basics of blockchain technology in section II. In section III, we discuss major vulnerabilities of blockchain technology and corresponding mitigation techniques for those vulnerabilities. Section IV introduces network level attacks on blockchain in the context of cryptocurrency applications. Section V concludes the paper with some future directions for further research on blockchain and its security.

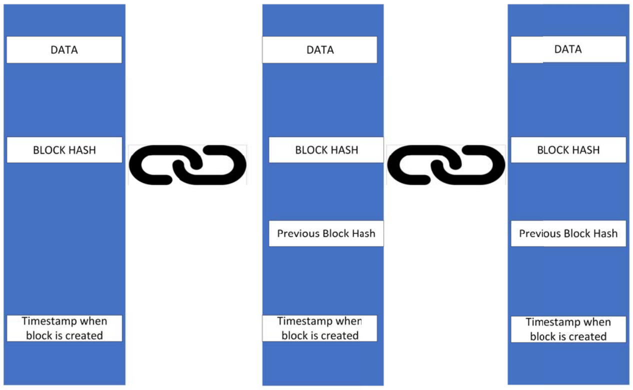
DATA
BLOCK HASH
Timestamp when block is created
DATA
BLOCK HASH
Previous Block Hash
Timestamp when block is created
DATA
BLOCK HASH
Previous Block Hash
Timestamp when block is created

Fig. 1. Blockchain structure.

II. BlockChain Basics

In a blockchain, data is stored in blocks, which are linked together in a chain using cryptographic hashes. Each block contains data, a hash of the previous block, a hash of the current block, and a timestamp of when the block was created. Because of the way the blocks are linked together, any attempt to change the data in one block would require changing all the subsequent blocks in the chain, making it nearly impossible to alter the blockchain without detection. To verify the integrity of a block efficiently, a Merkle tree based hash data structure is used. To add a new block to the blockchain maintained in a network, a consensus mechanism is used. All the nodes in the network must agree on the authenticity of the new block before it is added to the chain. This is done through a consensus-based agreement, in which the majority of nodes in the network must verify the new block before it is added. This consensus mechanism is designed to prevent dishonest attempts and malicious attacks, ensuring the integrity and security of the blockchain.

Blockchain's consensus mechanism, which enables many parties to concur on the state of the ledger without the need

979-8-3503-2759-5/23/$31.00 ©2023 IEEE
DOI 10.1109/CSCE60160.2023.00400

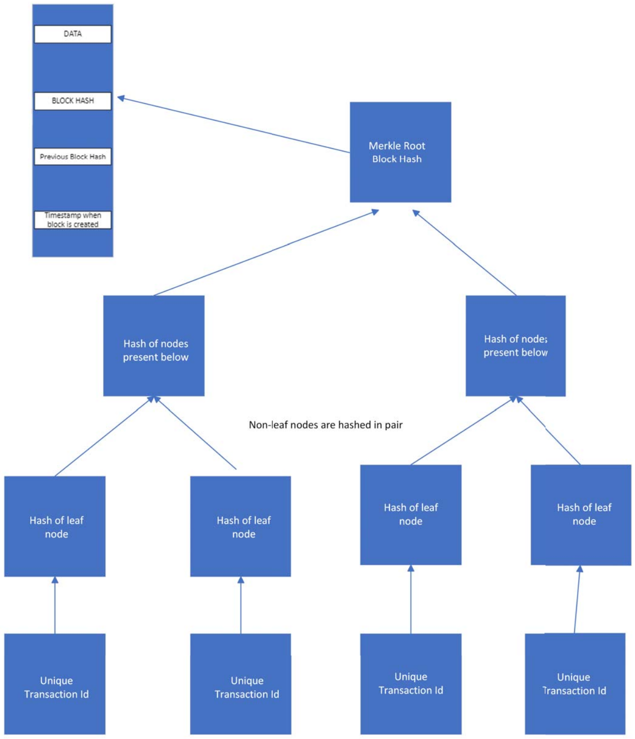

Fig. 2. Working of Merkle tree.

for a central authority, is one of its fundamental features. Numerous consensus algorithms, including proof-of-work (PoW), proof-of-stake (PoS), and delegated proof-of-stake, are used to accomplish this. In the following, some details on the PoW and the PoS mechanisms and their security vulnerabilities are discussed.

A. Merkle Tree

Many blockchain systems use a Merkle tree as a data structure to effectively store and verify large amounts of data. It is also referred to as a hash tree and bears Ralph Merkle's name, the name of its creator. Merkle trees are built by repeatedly hashing data pairs up until one root hash is produced [4]. Before all the data is hashed into a single root hash, each pair of data is first combined and hashed. The resulting hash is then combined with another pair of data, and so on. Without having to store or analyze the complete dataset, this format enables speedy and effective data integrity checking.

Each block in a blockchain system has a Merkle tree listing all the transactions that are part of that block. The blockchain can quickly confirm the legitimacy of a transaction without having to confirm all the other transactions on the network by hashing all the transactions together in a Merkle tree [5]. This enables the blockchain to handle transactions more quickly and effectively.

As depicted in Fig. 2, each leaf node in a Merkle tree represents the data being hashed, while each non-leaf node represents the hash of its child nodes. The leaf nodes of a Merkle tree in the context of a blockchain represent distinct transactions, while the tree's root corresponds to the root hash of all the transactions in a block. Each transaction in a blockchain system is given a transaction ID (TxID), which is a special identifier. The transaction data is typically hashed using a cryptographic hash function, like SHA-256, to generate the hash [6]. The Merkle root hash, which is the root hash of the Merkle tree of every transaction in the block, is also provided in the block header along with the TxID. The network can locate a transaction in the Merkle tree using the TxID in order to confirm the legitimacy of a transaction in a block rapidly and effectively by traversing the Merkle tree from the leaf node to the root node.

B. Proof Of Work

Many blockchain networks use the proof of work (PoW) consensus algorithm to validate transactions and add new blocks to the chain. Miners must solve a mathematical enigma called the nonce in order to validate a block and add it to the blockchain, which is a computationally demanding process. Finding a number that, when paired with the block data, creates a hash that satisfies specific requirements is the solution to the problem.

In the PoW algorithm, miners compete with one another to solve the puzzle, and the first one to do so is allowed to add the new block to the chain [11]. This procedure aims to safeguard the blockchain's security and immutability while preventing fraudulent transactions. However, PoW needs a lot of computing power, which can be expensive and energy-intensive. This raises concerns about its sustainability and environmental impact. In addition, it is vulnerable to attacks as discussed in the following section.

C. Proof Of Stake

In many blockchain networks, the proof of stake (PoS) consensus algorithm is an alternative to the proof of work consensus mechanism typically used in dealing with cryptocurrency. In PoS, validators are chosen based on the amount of cryptocurrency they hold and are willing to invest or lock up in order to participate in the consensus process, as opposed to miners competing to solve a mathematical puzzle [12]. The likelihood of getting selected as a validator and earning benefits increases with the stake level.

Transactions must be validated before being added to the chain, and new blocks must be added by validators. They run the danger of having their share reduced or lost if they approve fraudulent transactions or attempt to manipulate the system [13]. This procedure is intended to reward moral conduct and guarantee the blockchain's immutability and security. A committee of validators is in charge of proposing and validating blocks on the blockchain in proof of stake (PoS) consensus algorithms. The committee's size may change based on the particular PoS algorithm being employed. Some PoS systems choose the committee at random from the validators

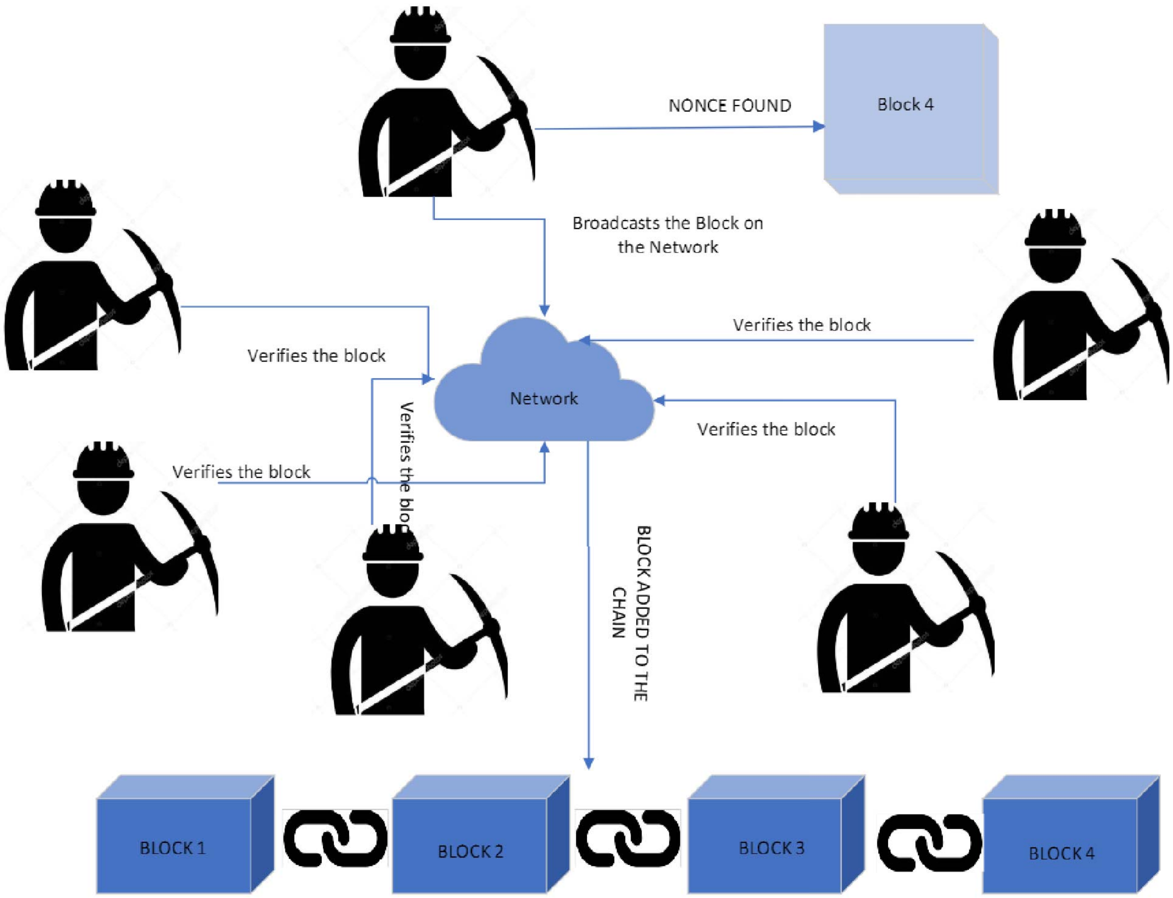

Fig. 3. Proof Of work.

in the pool based on various criteria, such as their stake or their standing within the network [14]. The members of the committee are in charge of suggesting new blocks and validating transactions within a given time frame.

The committee-based strategy aims to increase scalability and lower the amount of computing power needed for consensus in PoS systems. However, it imposes, to some extent, centralization and consolidation of power within a small number of significant validating nodes in the network.

In the following, we focus on major security vulnerabilities of blockchain technology, related smart contract technology, and recent attacks on various cryptocurrencies supported through blockchain networks [6], [15], [21].

III. Vulnerabilities in blockchain

A. 51-Percent Attack

The consensus mechanism of the blockchain is subject to 51 percent and double spending attacks [2], [20]. Because blockchain is based on a consensus method, a decision made by the majority of miners cannot be changed by any administrator. Blockchain technology is based on the consensus of the network's majority nodes. When the majority of nodes are compromised or taken over by an adversary, the network and transactions are jeopardized. The nodes in control can then reverse transactions, prevent new transactions from being confirmed, and prohibit some nodes from getting funds. All of this is only possible if the attacker has the complete control of the network.

After gaining control of the majority of the blockchain network, the attacker begins mining blocks in secret, without informing the remaining 49 percent of the chain. Parallel to the true chain, the attacker's secret chain runs on the network. The fraudulent chain is unknown to 49 percent of the nodes in the chain. Because the attacker has control over the majority of nodes, they begin mining blocks at a faster rate than the true chain. When the fraudulent chain outgrows the true chain, the nodes recognize the chain with the most work as the true chain, and all the legitimate nodes believe their own chain is incorrect

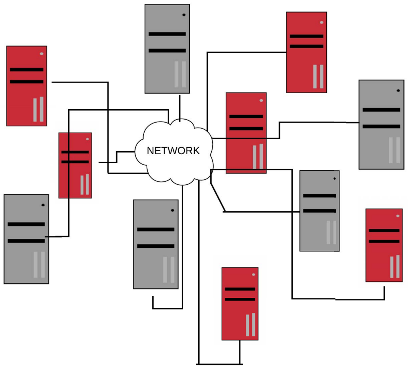

Fig. 4. 51 percent of compromised network nodes.

and accept the attacker's chain, according to the blockchain technology's longest chain rule [8]. Once the attacker's chain has been accepted into the network, they have complete control over the chain.

Even with the majority of nodes, the attacker will be unable to alter the locked transaction. Prior to an attack, all transactions are locked. The more transactions in the past, the more difficult it is to override them [9]. Even if the 51 percent attack succeeds, all validated transactions are locked into bitcoin's protocol, making modifications to those blocks impossible. A solution to withstand the 51 percent attack was offered by Komodo, an open multi-chain block platform. The Delayed proof of work (DPoW) is a hybrid consensus method that employs the hashpower of one blockchain to improve the security of a secondary blockchain. When a conflict occurs in the chain, DPoW looks for the most recent backups in the chain rather than the longest chain rule. The notary nodes perform a special hash every 10 minutes to keep track of the current height of the blockchain, and their digital signatures are also included in the notary data.When attempting to access the blockchain's history, even with the longest chain and hash rate, the DPoW mechanism will not transfer the right to a fake chain if it cannot be found in the most recent backup data [10]. Even if there is only one copy of the Komodo main chain, any attempt by the hacker will be overruled.

B. Double Spending

It is easy to duplicate and manipulate digital transactions due to technological flaws. As a result, it is critical to verify transactions before confirming and adding them to the blockchain. For example, if two people use the same account from a joint or family account. And if they do different transactions with the same money, it is referred to as double spending if there is insufficient money for both transactions. So, whenever a transaction is initiated, the system should check the balance and other factors to ensure that it is a legitimate transaction performed by the end user.

Attackers exploit this flaw to double spend the end user's digital currency by transferring money from the end user's wallet to the attacker's wallet, even if the end user's wallet does not have enough funds to do so. Attackers take advantage of this flaw to send money from one wallet to another out of thin air, as the victim's wallet does not need to have the funds to carry out the attack. In the case of digital currency, this attack results in counterfeit money, which causes inflation by creating a new currency that didn't exist before.

To successfully deploy this attack, attackers attempt to create a fork. When two conflicting transactions involving the same funds arrive at the nodes for verification, a fork is formed because the nodes cannot agree on which blocks to add. Once the fork is created, it is dependent on computational power to mine the nonce value faster, and whoever finds the nonce value for the transaction block will be added to the chain. When deploying this attack, attackers most likely take control of 51 percent of the nodes in order for the attack to be successful. Because once the fork is formed, it is all about which miner will find the nonce value of that transaction in order for it to be verified. With more computational power, an attacker will be able to find the nonce faster than other miners. All that is required for a transaction to be verified is for the miners to find the nonce. This creates a vulnerability for attackers to exploit because the node does not check whether the wallet has enough funds or whether the funds used are always updated in the wallet.To prevent these types of attacks, consensus mechanisms such as proof of work and proof of stake are to be implemented correctly and also there always should exist a backup of all the transactions. However, maintaining a backup of transactions in a distributed open environment is challenging as it requires further research on how to validate every transaction and how to ensure availability of the transactions reliably to all nodes in the network.

IV. Network Level Attacks

In this section, we present and analyze attacks on blockchain in the context of cryptocurrency applications such as bitcoin and etherium.

A. Transaction Malleability

The blockchain's ledger records every transaction that occurs on the network for future reference in the block chain. When a transaction is completed, it is given a transaction id and a block hash is generated to verify before adding it to the blockchain.

Transaction malleability is a network-level attack in which an attacker can tamper with a bitcoin's transaction id and introduce it into the network for verification. The attacker can claim that the initial transaction sent to him or her was never received. Unless checked and verified properly, he or she will receive an additional bitcoin payment thus doubling of what he or she was supposed to receive after the block is validated. Double spending and transaction malleability are two completely different types of attacks [16]. Double spending involves spending the coins once and then creating a new transaction with the same coins, causing the network to fork, however, transaction malleability involves changing the block's unique id and introducing it into the network, claiming that no payment was received, thus receiving the coins twice.

For instance, suppose Joshua runs an Ethereum exchange, and Lucifer is a potential user with X Ethereum coins on Joshua's platform. Lucifer now wants to withdraw funds from the exchange, so Joshua sends the funds to Lucifer, which creates a transaction, and Lucifer, with malice intent, can send the same transaction on the network by tweaking its transaction id whenever even a little bit of data is changed, the hash of the block changes, and the network does not detect it as a threat and accepts it.

To make this exploit work, Lucifer's duplicate block with a new transaction id must be validated on the network before Joshua's transaction, and after Lucifer's block is accepted on the network and he receives the Etherium coins for it, he claims that Joshua's transaction has not been received, and when Joshua checks the blockchain with the original transaction id and does not find it, he resends the coin to Lucifer. This transaction malleability can also occur when users use third-party software to manage their wallet [16]. To protect against transaction malleability attacks, transaction confirmations should be double-checked, and coin withdrawal exchanges should be manually verified [17]. Further research is needed how to automate detection of any malicious coin withdrawal.

B. *Eclipse Attack*

Eclipse attacks and Sybil network attacks appear to be similar [19], but a Sybil attack focuses on gaining control of all nodes in the network and controlling the nodes, whereas an eclipse attack focuses on isolating a single node from the network and manipulating its decisions [18].

Due to bandwidth constraints, nodes in a decentralized network cannot communicate with all other nodes in the network, and nodes can only connect up to 125 nodes, for example (depending on the chain). The attacker bombards the user with a large number of IP addresses that the attacker controls and with which the target node may connect in the future. Either the attacker waits patiently for malicious nodes to connect as neighbors, or the attacker launches a distributed denial of service attack, forcing the target to restart the service [18]. When the target restarts or starts the crypto software, it attempts to connect with neighbor nodes, which is when the attacker tries to connect a corrupt node with the target or victime node, isolating it from the network. The victim node may be unaware that it is connected to malicious nodes and starts participating in the network as usual, assuming that nothing has been compromised.

The consequences of the eclipse attacks are:

1) Double Spending - Once the target has been removed from the network, the attacker manipulates the node into accepting a transaction created by a user in order to double spend the coins [18].
2) Miner power disruption - The attacker takes control of several network nodes, isolates them from the blockchain, and forces them to mine for orphan blocks. Orphan blocks are nodes that are removed from the blockchain when two miners discover a block at the same time and broadcast it to the network. The attacker's chances of finding the nonce will increase as the nodes mine for orphan blocks, allowing them to earn block rewards [19].
3) If the attacker can isolate enough nodes from the network to gain 51 percent of the chain, they can launch a 51-percent attack [2].

To protect against eclipse attacks, the exchange platform can use a strict policy on inbound connections such as blocking inbound connections and only connecting to outbound connections with specific nodes in the network that are trusted by other nodes [19]. However, new nodes will find it difficult to join the network if this is implemented. In this case, increasing the number of node connections in the network can help to mitigate these attacks as it increases the probability of a legitimate node to connect to a node [19]. However, the solution still leaves nodes vulnerable to eclipse attacks, and hence, further research is needed to better protect nodes from such attacks.

V. Recent Blockchain Hacks

A. *Coinbase Hack*

Between March and May 2021, Coinbase, a leading cryptocurrency platform, experienced a security breach, resulting in the loss of cash for at least 6,000 clients [22]. According to Coinbase's report, hackers used phishing tactics to get access to clients' email addresses, passwords, and phone numbers associated with their accounts. According to Coinbase, there was no proof that the information was obtained from the Coinbase itself. Threat actors used phishing tactics to acquire access to the user's information. Users have received emails claiming that their Coinbase account has been locked and that they must enter credentials such as their email address and password to reactivate it. They also sent an email asking the customer to allow Coinbase to view their personal email [23]. This is how the hackers gained access to the information they needed. However, the hacker still needed to go past the multifactor authentication process.

Every user on Coinbase must set up authentication methods such as based on SMS, Timed One-Time Password, and Hardware Device 2 Factor Authentication [22]. However, the accounts compromised were those that employed SMS as a multi-factor authentication method, since there was a weakness in the SMS account recovery process to obtain a token and gain access to the account. Though the coinbase did not provide an explanation on the SMS authentication problem, it was speculated that the hackers gained access by implementing the SIM swap attack [23].

SIM switch attacks are a type of phishing attack in which the attacker uses social engineering to acquire access to the victim's phone number [24]. After gaining access to the victim's personal information, the hackers enter the relevant

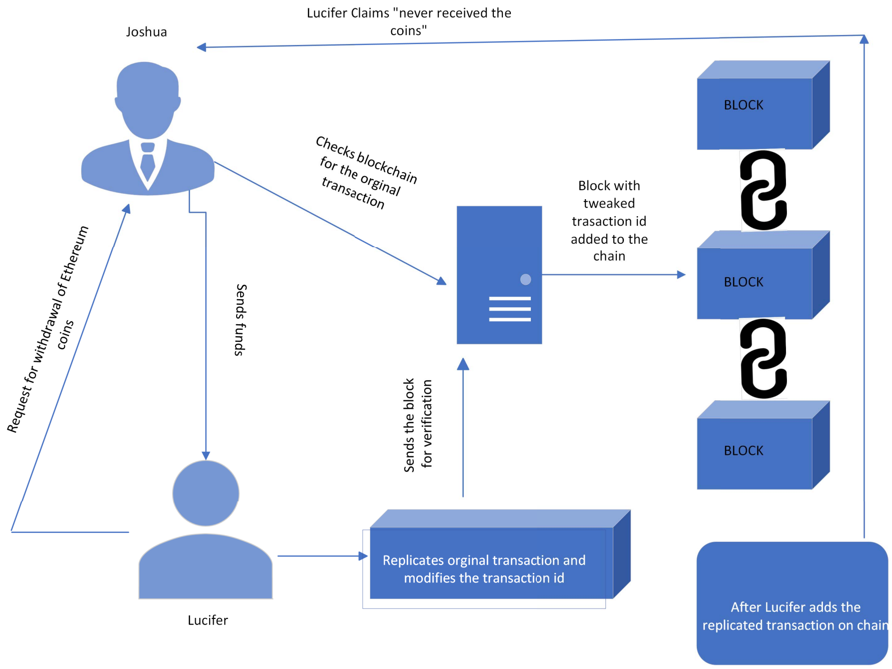
Lucifer Claims "never received the coins"
Joshua
Checks blockchain for the orginal transaction
BLOCK
Block with tweaked trasaction id added to the chain
BLOCK
BLOCK
Request for withdrawal of Ethereum coins
Sends funds
Sends the block for verification
Lucifer
Replicates orginal transaction and modifies the transaction id
After Lucifer adds the replicated transaction on chain

Fig. 5. Transaction malleability attack.

information on a carrier website or telephone customer care to persuade them that they are the user and that they want their phone number remotely switched to a threat actor's SIM card. This is how hackers can circumvent the SMS authentication process and steal payments from consumers by switching ports. Coinbase has indicated that the potential weakness has been resolved but without giving any other information to its users [23]. Customers should utilize a stronger mechanism to secure their accounts, such as a time-based one-time password or a hardware security key.

B. Poly Network Hack

Poly network is a cross-chain platform that supports different blockchains, including bitcoin, ethereum, and the Binance smart chain. It is a mechanism for transferring tokens from one digital ledger to another for heterogeneous blockchains. A cross chain network allows two blockchains to communicate with one another [25]. Due to flaws in two smart contracts on the poly network, the poly network hack was possible. The tokens were transferred to several cryptocurrency wallets after a hacker exploited a vulnerability between contract calls in the system. On August 10th, 2021, an unknown hacker or hackers launched an attack that resulted in the theft of approximately 610 million US dollars worth of tokens and transferred them to a hacker-controlled address.

The EthCrossChainManager contract, which is in charge of the network's cross-chain transactions, is one of the contracts that has been exploited. By calling the method VerifyHeaderAndExecuteTX, it verifies blocks and adds them to the chain [25]. The EthCrossChainManager can call the contract EthCrossChainData, which is the major flaw. EthCrossChainManager controls access to funds by maintaining a list of keepers, while EthCrossChainData allows its owner to update the wallet's keepers [26]. EthCrossChainData is a privileged contract that is only supposed to be used by its owners because it is in charge of managing the wallets' public keys. If EthCrossChainData is tampered with, it can offer the hacker access to the network's funds and allow them to be transferred to other wallets.

The attacker can acquire control of the EthCrossChainManager and manage the attacker's public key as the keeper of the wallet and transfer tokens in and out of the chain by brute forcing the 32-bit value solidity function id [25]. Thus, the hacker can mislead the EthCrossChainManager into invoking the EthCrossChainData and executing privileged instructions within the function call. Poly Network ordered all crypto chain platforms to freeze the stolen assets, rendering them unavailable to the hacker, as a move to control the attack and retrieve the lost funds. Poly Network sent out a tweet urging the hacker to return the funds with an offer of a 500,000 dollars bounty and an invitation to become its chief security advisor. The hacker then slowly refunded the payments, explaining that it was done to demonstrate the poly network's security weaknesses. The case demonstrates that a better solution for securely transferring tokens from one ledger to another needs to be found and implemented properly for all blockchain cross platform operations.

C. Ethereum Client Geth Vulnerability

The ethereum node's command line interface, Geth written in the Go programming language, had a security flaw. To mitigate the security flaw, the blockchain for Ethereum was split into two chains by carrying out a hard fork [27]. In August, 2021, Ethereum posted a hotfix for the security flaw and asked all clients to update their software [27]. When Ethereum acknowledged the security weakness, several hackers gradually discovered the flaw and began exploiting it, stealing funds from users still on previous versions. Only 38 percent of nodes updated once the hotfix was released, and all earlier version nodes were exposed to the exploit [28]. Eventually, Ethereum had to go through a hard fork in order to confirm all of the transactions and return the funds to the affected users. The case demonstrates that any inherent security flaws in existing software systems used for implementation of a blochchain can cause security problems, and it is imperative to check the software products before using them to build a blockchain network and services.

VI. Conclusion and Future Scope

In this work, we present an analysis of major security vulnerabilities, attacks, and advances in blockchain technology. The decentralized network of blockchain technology helps to mitigate a cyber attack directed at a single point because the entire system does not go down and continues to operate independently. However, one compromised node can put the data at risk because attackers will have access to all of the information stored on the network's ledger. As a result, it is critical to examine the impact of these risks on technology, as well as whether it can withstand breaches and keep data safe. More research into probable security breaches and solutions to limit the risk associated with blockchain technology should be conducted to make the technology dependable.

A decentralized network's storage is also a drawback. Because every node has an identical copy of the data, storing it and transferring it takes a lot of computing power, and the time it takes to process a chain grows over time. Accordingly, data sharding optimization techniques should be the subject of more intensive research.

Ethereum platform, bitcoin, and smart contracts are widely used in the blockchain industry and they need to be secured as they contain all the digital transactions. More study can be conducted on the threats pertaining to them and also review various methods in which blockchain platforms can be used to develop solutions for the cyber security threats. Machine learning techniques such as unsupervised learning, which is utilized for anomaly detection, can also be used to monitor network entry. These traffic monitors can send out alerts to clients if there is anything unusual going on in the chain. More research in algorithms is needed to filter fraudulent traffic and warn users about the danger.

References

[1] Yli-Huumo J., Ko D., Choi S., Park S., and Smolander K., "Where is current research on blockchain technology?—a systematic review. PLoS ONE," vol. 11, no. 10, 2016. https://doi.org/10.1371/journal.pone.0163477

[2] Hasanova, H., Baek, U.-J., Shin, M.-G., Cho, K., and Kim, M.-S., "A survey on blockchain cybersecurity vulnerabilities and possible countermeasures," International Journal of Network Management, vol. 29, no. 2, 2019.

[3] Saini, H., Bhushan, B., Arora, A., and Kaur, A., "Security vulnerabilities in information communication technology: blockchain to the rescue (a survey on blockchain technology)," In 2019 2nd IEEE International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), Vol. 1, pp. 1680-1684, July 2019.

[4] Lin, I.-C. and Liao, T.-C., "A survey of blockchain security issues and challenges," International Journal of Network Secururity, vol. 19, no. 5, pp. 653-659, 2017.

[5] Zheng, Z., et al., "Blockchain challenges and opportunities: A survey," International Journal of Web and Grid Services, vol. 14, no. 4, pp. 352-375, 2018.

[6] Lin, I. and Liao, T., "A Survey of blockchain security issues and challenges," International Journal of Network Security, vol. 19, pp. 653-659, 2017.

[7] Huynh-The, T., et al., "Blockchain for the metaverse: a review," Future Generation Computer Systems, 2023.

[8] Yaga, D., et al., "Blockchain technology overview," arXiv preprint, 2019, arXiv:1906.11078.

[9] Bashir, I., "Mastering blockchain," Packt Publishing Ltd, 2017.

[10] Iansiti, M. and Lakhani, K. R., "The truth about blockchain" Harvard Business Review, vol. 95, no. 1, pp. 118-127, 2017.

[11] Dasgupta, D., Shrein, J. M., and Gupta, K. D., "A survey of blockchain from security perspective," Journal of Banking and Financial Technology, vol. 3, pp. 1-17, 2019.

[12] Saleh, F., "Blockchain without waste: Proof-of-stake" The Review of financial Studies, vol. 34, no. 3, pp. 1156-1190, 2021.

[13] Li, W., et al., "Securing proof-of-stake blockchain protocols," Data Privacy Management, Cryptocurrencies and Blockchain Technology: ESORICS 2017 International Workshops, Oslo, Norway, September 14-15, 2017.

[14] Kiayias, A., et al., "Ouroboros: a provably secure proof-of-stake blockchain protocol," Advances in Cryptology–CRYPTO 2017: the 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20–24, 2017.

[15] Amiet, N., "Blockchain vulnerabilities in practice," Digital Threats: Research and Practice, vol. 2, no. 2, pp. 1-7, 2021.

[16] Averin, A. and Averina, O, "Review of blockchain technology vulnerabilities and blockchain-system attacks," The IEEE 2019 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon), 2019.

[17] Singh, A., et al., "Blockchain smart contracts formalization: approaches and challenges to address vulnerabilities," Computers Security, vol. 88, 2020.

[18] Kushwaha, S. S., et al., "Systematic review of security vulnerabilities in ethereum blockchain smart contract," IEEE Access, vol. 10, pp. 6605-6621, 2022.

[19] Dika, A. and Nowostawski, M., "Security vulnerabilities in ethereum smart contracts," The 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2018.

[20] Rebello, G. A. F., et al., "A security and performance analysis of proof-based consensus protocols," Annals of Telecommunications, pp. 1-21, 2021.

[21] Chen, H., et al., "A survey on ethereum systems security: Vulnerabilities, attacks, and defenses," ACM Computing Surveys (CSUR), vol. 53, no. 3, pp. 1-43, 2020.

[22] Hasanova, H., et al., "A survey on blockchain cybersecurity vulnerabilities and possible countermeasures," International Journal of Network Management, vol. 29, no. 2, 2019.

[23] Khangura, J. and Arora, J., "A Study on security threats to blockchain cryptocurrencies," The IEEE 3rd International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), 2021.

[24] Asim, J., et al., "Blockchain-based multifactor authentication for future 6G cellular networks: a systematic review," Applied Sciences, vol. 12, no. 7, 2022.

[25] Kang, I., Gupta, A., and Seneviratne, O., "Blockchain interoperability landscape," arXiv preprint, 2022, arXiv:2212.09227.

[26] He, D., et al. "Detection of vulnerabilities of blockchain smart contracts," IEEE Internet of Things Journal, 2023.

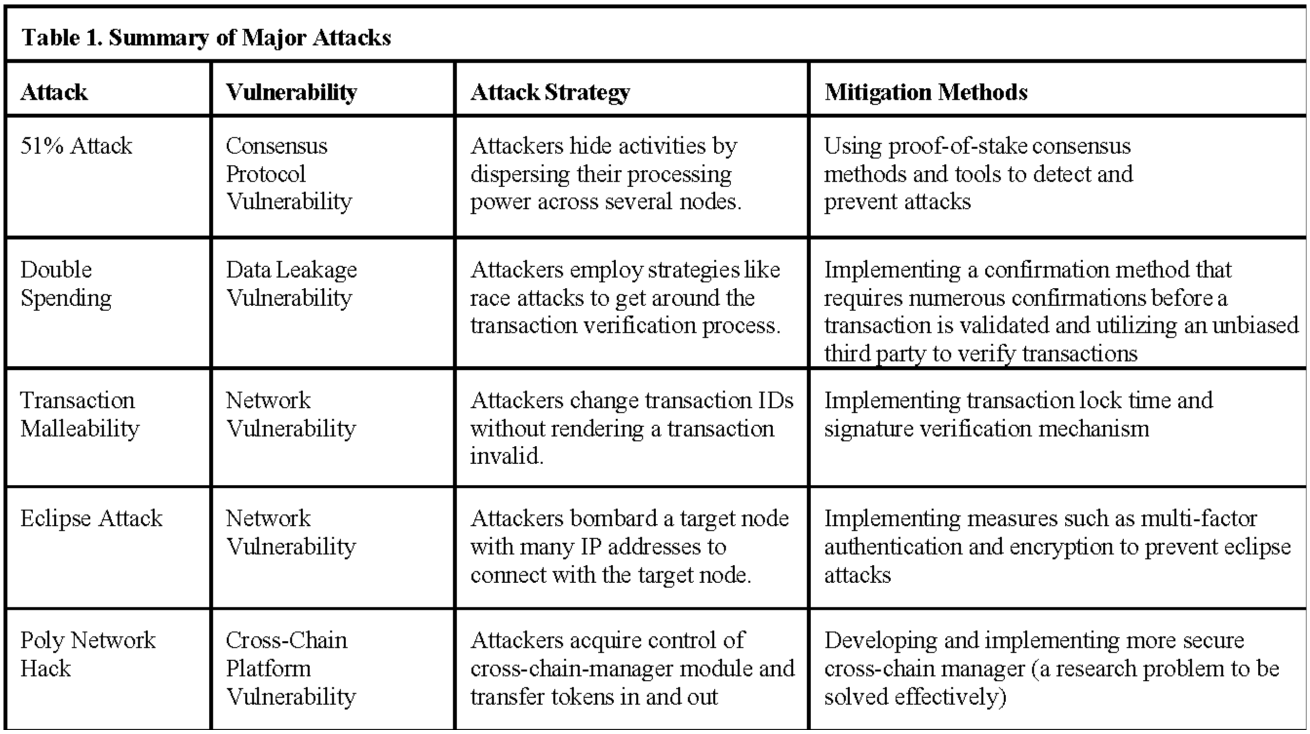

Table 1. Summary of Major Attacks

Attack	**Vulnerability**	**Attack Strategy**	**Mitigation Methods**
51% Attack	Consensus Protocol Vulnerability	Attackers hide activities by dispersing their processing power across several nodes.	Using proof-of-stake consensus methods and tools to detect and prevent attacks
Double Spending	Data Leakage Vulnerability	Attackers employ strategies like race attacks to get around the transaction verification process.	Implementing a confirmation method that requires numerous confirmations before a transaction is validated and utilizing an unbiased third party to verify transactions
Transaction Malleability	Network Vulnerability	Attackers change transaction IDs without rendering a transaction invalid.	Implementing transaction lock time and signature verification mechanism
Eclipse Attack	Network Vulnerability	Attackers bombard a target node with many IP addresses to connect with the target node.	Implementing measures such as multi-factor authentication and encryption to prevent eclipse attacks
Poly Network Hack	Cross-Chain Platform Vulnerability	Attackers acquire control of cross-chain-manager module and transfer tokens in and out	Developing and implementing more secure cross-chain manager (a research problem to be solved effectively)

[27] Kissoon, Y. and Bekaroo, G., “Detecting vulnerabilities in smart contract within blockchain: a review and comparative analysis of key approaches,” The IEEE 3rd International Conference on Next Generation Computing Applications (NextComp), 2022.

[28] Nguyen, D. C., et al., “Bedgehealth: a decentralized architecture for edge-based iomt networks using blockchain,” IEEE Internet of Things Journal, vol. 8, no. 14, pp. 11743-11757, 2021.