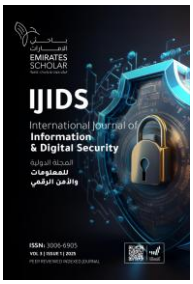

International Journal of Information & Digital Security
Vol 3 Issue 1 (2025)
Pages (4 - 17)

Available at
www.emiratesscholar.com

The Role of Cryptography in Securing Blockchain Networks

Saher Hassan[1], Mohamed Abdallatif[2], Mahmoud Atia[3]
Computer Engineering, Benha University , Shoubra, Shoubra, 13511, Qalyubia, Egypt[1,2,3]
sahar.hassan@feng.bu.edu.eg[1], mohamedtofa21@gmail.com[2], mahmoudmohamed5264@gmail.com[3]

ARTICLE HISTORY

Received: 19 November 2025.
Accepted: 30 November 2025.
Published: 24 December 2025.

PEER - REVIEW STATEMENT:

This article was reviewed under a double-blind process by three independent reviewers.

HOW TO CITE

Hassan, S., Abdallatif, M., & Atia, M. (2025). The Role of Cryptography in Securing Blockchain Networks. *International Journal of Information & Digital Security*, *3*(1), 4-17. https://doi.org/10.54878/q0bzep24

ABSTRACT

Blockchain technology is a game-changing invention that guarantees digital transactions on decentralized networks. The vital role that cryptography plays in guaranteeing the authenticity, confidentiality, and integrity of blockchains is examined in this paper. To secure the data on the blockchain and validate transactions, we are examining fundamental cryptographic techniques like hashing, symmetric and asymmetric encryption, and digital signatures. Furthermore, advanced cryptographic solutions that have the potential to improve privacy and scalability—such as homomorphic encryption, zero-knowledge proofs, and zk-SNARKs—are being discussed. Along with reviewing consensus techniques like proof of work and proof of stake, the paper contrasts the main blockchains, including those that are still in development, like Ethereum, Solana, and Hyperledger Fabric. Through an analysis of the advantages and disadvantages of existing cryptographic implementations, the study emphasizes the necessity for additional innovation.

Keywords: *Blockchain, Blockchain security, Consensus algorithm*

1. Introduction

Satoshi Nakamoto's seminal 2008 paper, "Bitcoin: A Peer-to-Peer Electronic Cash System", presented the concept of Bitcoin, a decentralized digital currency underpinned by blockchain technology. This technology employs a chronologically linked data structure of blocks, utilizing cryptographic methods to guarantee data integrity, traceability, and security, thus creating a tamper-proof and forgery-resistant record.

As blockchain technology matures, the demand for enhanced data protection, anonymity, and untrace ability across diverse applications is increasing [1]. Initially conceived for cryptocurrencies [2], blockchain has expanded its reach into sectors like healthcare, intellectual property, and finance, and is recognized as a leading solution for ensuring personal data privacy and secure information exchange [3]. The growth of big data and the proliferation of Internet of Things (IoT) devices have further underscored the critical need for secure data sharing, a domain where blockchain plays a crucial role. Addressing internal security risks stemming from trust deficits among users, blockchain offers a decentralized framework for establishing trust without the need for central authorities.

However, the broader adoption of blockchain has introduced security challenges, particularly concerning privacy and transaction safety. Moreover, the progress in quantum computing presents a significant future threat to the cryptographic foundations of current blockchain systems [4]. Consequently, various cryptographic techniques are under development to bolster blockchain security.

This paper aims to explore the cryptographic methods integral to blockchain systems. To provide a thorough understanding, we will first outline the fundamental principles and architecture of blockchain technology, followed by an introduction to the primary types and categories of cryptographic protocols.

2. Background

At its core, blockchain functions as a decentralized database. It is built upon three essential components: distributed ledger technology, immutable records, and smart contracts. Distributed ledger technology enables all participants within a blockchain network to access a shared ledger that contains a complete and unchangeable history of transactions [5]. This approach ensures that each transaction is recorded only once, thereby eliminating redundancies commonly found in traditional business systems. The immutability of records means that once a transaction is entered into the blockchain, it cannot be altered by any participant. If an error occurs in a transaction, it cannot be edited directly; instead, a new corrective transaction must be added. Both the original and corrective transactions remain visible in the ledger, ensuring transparency and traceability. Smart contracts are predefined sets of rules stored on the blockchain that are automatically executed when certain conditions are met [6]. These contracts help streamline and automate transaction processes, reducing the need for intermediaries.

As illustrated in Figure 1, each block in a blockchain consists of two main parts: the block header and the block body. The block header contains critical metadata, including the block version number, the hash of the previous block, a Merkle tree root, a nonce, and a timestamp. The Merkle tree root is a single hash value that efficiently summarizes all the transactions in the block, allowing for quick and secure verification of transaction integrity. The nonce is a randomly generated number that miners adjust during the mining process to solve the cryptographic puzzle required for block validation. In contrast, the block body stores the actual transaction data relevant to that specific block.

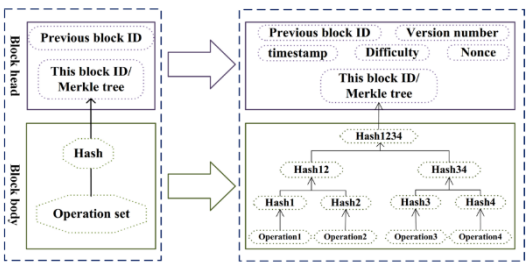

Figure. 1. Block structure [7].

Blockchain technology's robust framework is built upon several cryptographic concepts that work in concert to ensure its security, integrity, and functionality. These fundamental cryptographic pillars include hashing algorithms, asymmetric cryptography (involving public and private keys), digital signatures, and, to a lesser extent, symmetric cryptography.

2.1 Hashing Algorithms

At the core of blockchain architecture is the utilization of cryptographic hash functions. These functions process inputs of arbitrary size and generate fixed-size outputs, commonly referred to as a hash or message digest. This output serves as a unique digital fingerprint of the original data [8]. Several fundamental properties of cryptographic hash functions are particularly critical to their role in blockchain systems:

- **Preimage resistance**: Given a specific hash value, it should be computationally infeasible to determine the original input that produced it [9]. This property guarantees that reversing the hashing process to retrieve the original data is practically impossible.
- **Second preimage resistance**: For a given input, it should be computationally infeasible to find another distinct input that produces the same hash output [9]. This ensures that an attacker cannot replace one piece of data with another while preserving the same hash.
- **Collision resistance**: It must be computationally infeasible to identify two different inputs that result in the same hash output. Although collisions may theoretically occur due to the fixed output size and variable input size, strong cryptographic hash functions render the probability of finding such collisions negligibly small.

A range of widely adopted cryptographic hash algorithms are employed in blockchain systems. SHA-256 (Secure Hash Algorithm 256-bit) is notably utilized in Bitcoin [8], while Ethereum implements a modified version of SHA-3, known as Keccak-256. More recently, BLAKE3 has attracted attention due to its enhancements in speed and security properties.

Hashing plays a pivotal role in establishing the structural integrity of a blockchain. Each block's header—except for the initial genesis block—contains the cryptographic hash of the preceding block's header [8]. This mechanism creates an immutable chain of blocks, where each block is securely linked to its predecessor. Any modification to the data within a previous block would result in a different hash output, thereby disrupting the chain and making tampering immediately detectable [8].

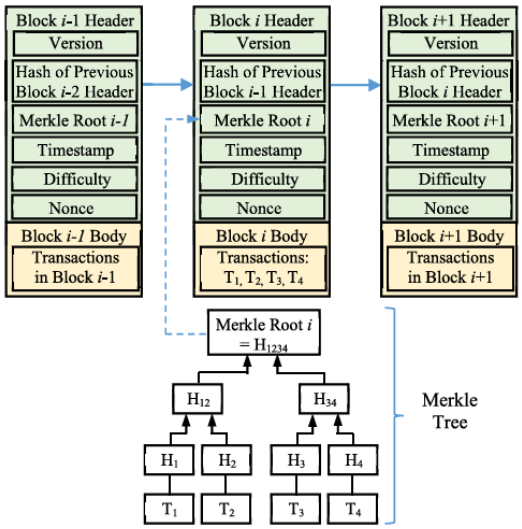

Figure. 2. Blockchain connection structure and a Merkle tree with hash function[10].

2.2 Asymmetric Cryptography (Public and Private Keys)

Asymmetric cryptography, also referred to as public-key cryptography, constitutes another fundamental pillar of blockchain security. This cryptographic scheme employs a pair of mathematically associated keys for each participant: a public key and a private key [10]. The public key is intended for broad dissemination and is utilized for encryption and for verifying digital signatures, whereas the private key is confidentially maintained by its owner and is used

for decryption and for the creation of digital signatures [10]. Although the two keys are mathematically related, deriving the private key from the public key is computationally infeasible, thereby ensuring the security of the system [9].

Within blockchain systems, public keys frequently function as user identifiers or wallet addresses for the receipt of digital assets. When a transaction is initiated, the sender typically designates the recipient's public key as the destination address. Subsequently, the corresponding private key is employed by the recipient to authorize and validate transactions, such as spending or transferring the received cryptocurrency [8].

2,3 Digital Signatures

Digital signatures represent a crucial application of asymmetric cryptography within blockchain systems, serving as a fundamental mechanism for transaction authentication and non-repudiation [8]. When a user initiates a transaction on a blockchain network, they employ their private key to generate a digital signature for the transaction. This digital signature is created by producing a cryptographic hash of the transaction data and subsequently encrypting the hash with the sender's private key [8].

Verification of the digital signature can be performed by any party with access to the sender's corresponding public key [8]. The verification process involves decrypting the digital signature using the public key and comparing the obtained hash with freshly computed hash of the original transaction data. A match between the two hashes provides cryptographic assurance that the transaction was indeed signed by the legitimate owner of the associated private key and that the transaction data has remained unaltered since signing [8]. This process guarantees both the authenticity of the sender and the integrity of the transaction while also providing non-repudiation, meaning that the sender cannot plausibly deny authorizing the transaction [9].

Digital signatures thus play a pivotal role in securing a wide range of blockchain-based applications, including cryptocurrency transactions, the maintenance of supply chain integrity, the development of secure electronic voting systems, and the execution of smart contracts [9].

2.4 Symmetric Cryptography

Symmetric cryptography employs a single secret key for both the encryption and decryption of data [9]. Although asymmetric cryptography underpins essential blockchain operations such as authentication and digital signatures, symmetric cryptography is generally less prevalent in core blockchain functions, particularly transaction signing, due to the inherent difficulties in securely distributing a shared secret key across all participants in a permissionless network. Nevertheless, symmetric encryption can serve important roles within blockchain systems, particularly for encrypting the contents of transactions in permissioned blockchain networks. Advanced Encryption Standard (AES) is a prominent example of a widely adopted symmetric cipher used in such contexts [9].

The integration of these cryptographic techniques forms robust security architecture underlying blockchain technology. Hashing ensures the integrity and linkage of blocks; asymmetric cryptography enables secure identity management and transaction authorization; and digital signatures provide authentication and non-repudiation, collectively safeguarding the trustworthiness and resilience of blockchain-based systems.

3. Literature Review

Blockchain technology establishes trust among unrelated parties, enabling secure and verifiable records and transactions without the need for traditional intermediaries. By leveraging cryptography and collaborative validation, blockchain creates this trust, effectively removing the reliance on central authorities. Cryptographically secured information is recorded on its distributed ledger. Blockchain utilizes cryptographic tools such as public-key cryptography for digital signatures and encryption, zero-knowledge proofs for

demonstrating knowledge without disclosure, and hash functions, which are also integral to Merkle trees in block headers.

3.1 Blockchain consensus algorithms:

3.1.1 Proof of Work(PoW):

The concept of Proof of Work was introduced before Bitcoin, with early implementations found in **Hashcash by Adam Back (1997)** who Introduced the idea of using computational puzzles to limit email spam and who adapted PoW for decentralized consensus in blockchain. Cryptocurrencies like Litecoin, and Bitcoin are currently using PoW [11]. Ethereum was using PoW mechanism but now shifted to Proof of Stake(PoS). The process of solving the crypto puzzle is called mining [12]. The mining activity performed by the node in the network is named as minor. Then the new block is added to the blockchain as shown in Figure. 3.

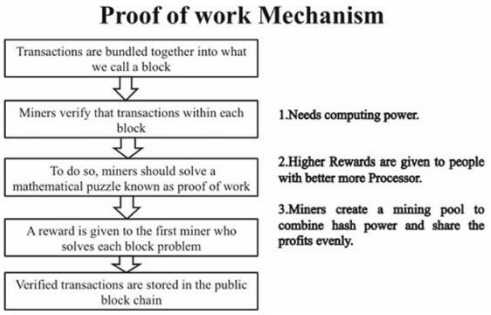

Figure. 3. Flow of Proof of Work [12].

Problems of PoW:

1. **51% Attack Vulnerability**
    - While improbable for major cryptocurrencies like Bitcoin, there's a risk of a 51% attack where a malicious entity gains control of the majority of the network's hashing power [11].
    - This control allows for fraudulent transactions, double-spending, and manipulation of the blockchain's history.
    - Smaller, less valuable cryptocurrencies are more susceptible to this attack.
2. **Time-Consuming Process:**
    - Solving the cryptographic puzzle to mine a block requires miners to exhaustively search for the correct solution making block creation slow.
3. **High Resource Consumption:** The process demands significant computational power, leading to the wastage of hardware, physical space, financial investment, and substantial energy.

3.1.2 Proof of Stake(PoS):

Proof of Stake (PoS) offers an alternative to the Proof of Work (PoW) consensus mechanism while maintaining similar goals for cryptocurrency cybersecurity. Like Proof of Work, Proof of Stake relies on a blockchain as a digital ledger to record and validate transactions involving digital assets. In Proof of Stake, the selection of the next block creator is determined by the amount of cryptocurrency a user chooses to "stake," essentially locking up their investment[11]. To ensure network security, Proof of Stake implements a penalty system where validators who attempt fraudulent activities **risk losing a portion** or all their staked cryptocurrency. ·While validators are often randomly selected, the probability of selection is weighted by the amount of cryptocurrency staked, giving those with **larger stakes a higher chance**, akin to having more "votes". Unlike Proof of Work, Proof of Stake does not involve a **competitive** process where participants solve complex computational puzzles,[12] and there are no direct rewards for this type of work. Validators in Proof of Stake have a strong financial incentive to act honestly, as any validation or initiation of fraudulent transactions can lead to the loss of their staked assets and being barred from future validation.

Instead of direct block rewards from puzzle-solving, Proof of Stake participants are typically rewarded through network transaction fees and

the creation of new tokens, distributed based on their staked amount. To become a validator in a Proof of Stake system, users generally need to stake a specific minimum quantity of the cryptocurrency, such as the 32 ETH required by Ethereum (as of November 19, 2022 is worth approximately $38K USD) [12]. Proof of Stake fundamentally changes the energy consumption profile of verifying blockchain transactions by shifting the workload from energy-intensive computations to the hardware of users who are actively staking their cryptocurrency. The following figure indicates how proof of stake works.

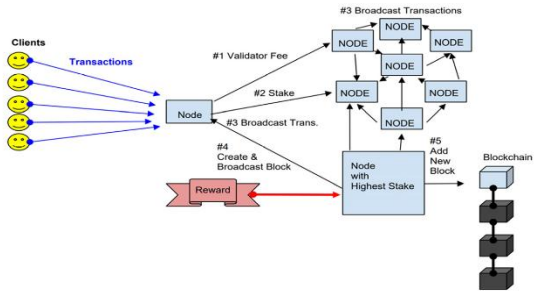

Figure. 4 Flow of Proof of Stake [12]

Security concerns of PoS:

- Fixed Coin Supply: PoS cryptocurrencies typically have a fixed, finite number of coins in circulation, with no new coin creation after the initial distribution or a transition from PoW.
- Forger Rewards and Penalties: The validator ("forger") creating a new block receives transaction fees, but they lose these fees and a larger portion of their stake if the block is proven fraudulent.
- 51% Attack Impracticality: A 51% attack is considered a poor security risk due to the immense cost and time required to acquire such a large stake, making it unprofitable.

Table. 1 Comparison between PoW and PoS.

Feature	Proof of Work (PoW)	Proof of Stake (PoS)
Validation Work	Requires significant computational work (energy-intensive).	Requires significantly less work (more energy-efficient).
Competition	Large number of miners compete to solve a puzzle.	Selection process (random draw) rather than direct competition.
Selection	First to solve a complex mathematical puzzle validates.	Selected based on stake, age of stake/coins, and randomness.
Reward	Miner receives a block reward (newly minted coins, tokens).	Validator receives transaction fees and potentially block rewards.
Block Creation	Blocks are "mined" through computational effort.	Blocks are "forged" by selected validators.
Initial Coin State	Coins are typically created as mining rewards.	Coins can be pre-mined or the system can transition from PoW.
Validator Incentive	Reward for successfully mining a block.	Ability to collect transaction fees and potential block rewards; risk of losing stake for dishonesty.

3.2 Blockchain platforms

3.2.1 Ethereum:

Native Cryptocurrency: Ether (ETH)

Key Features: Pioneered smart contract functionality, has a large and active developer community, and supports a vast ecosystem of dApps and DeFi (Decentralized Finance) projects. It is currently transitioning to Ethereum 2.0, which involves a shift to a Proof-of-Stake consensus mechanism to improve scalability and energy efficiency[13].

Use Cases: DeFi applications, NFTs (Non-Fungible Tokens), DAOs (Decentralized Autonomous Organizations), and a wide range of other decentralized applications.

3.2.2 Solana:

Native Cryptocurrency: SOL

Key Features: Designed for high speed and low transaction costs, utilizing a unique hybrid consensus mechanism combining Proof-of-History (PoH) and Proof-of-Stake. It aims to provide a scalable platform for dApps [13].

Use Cases: High-frequency trading platforms, decentralized exchanges, and applications requiring fast transaction processing.

3.2.3 Hyperledger Fabric:

Native Cryptocurrency: None (permissioned blockchain)

Key Features: An open-source, permissioned blockchain framework hosted by the Linux Foundation. It is designed for enterprise use, focusing on privacy, scalability, and modularity. It supports smart contracts written in Go, Java, and Node.js [13].

Use Cases: Supply chain management, asset tracking, and other enterprise-level applications requiring a private and controlled blockchain environment.

3.2.4 R3 Corda: Native Cryptocurrency: None (permissioned blockchain)

Key Features: Another open-source, permissioned blockchain platform designed for businesses, particularly in the financial services industry. It focuses on direct, peer-to-peer transactions with restricted data sharing [14].

Use Cases: Financial transactions, supply chain finance, and digital identity.

3.2.5 Tezos:

Native Cryptocurrency: XTZ

Key Features: Known for its on-chain governance mechanism, allowing the protocol to upgrade itself without hard forks. It supports smart contracts with formal verification, enhancing security. It uses a Liquid Proof-of-Stake (LPoS) consensus mechanism [13].

Use Cases: Applications requiring high levels of security and formal verification, and those benefiting from on-chain governance.

3.2.6 Stellar:

Native Cryptocurrency: XLM (Lumens)

Key Features: Focuses on facilitating cross-border payments and the tokenization of assets. It has a fast and low-cost network with a unique consensus protocol called the Stellar Consensus Protocol (SCP) [14].

Use Cases: International money transfers, micro-payments, and asset tokenization.

Table 1. Blockchain platforms comparison

Platform	Github Package Name	Permissionless	Consensus	Cryptocurrency
Ethereum	ethereum/go-ethereum	Yes	PoS	$ETH (Ether)
Solana	solana-labs/solana	Yes	PoH (Similar to PoS)	$SOL (Solana)
Hyperledger Fabric	hyperledger/fabric	No	Kafka, Solo, BFT-SMaRt	-
R3 Corda	corda/corda	No	Notary Nodes	-
Tezos	tezos/tezos	Yes	PoS	$XTZ (Tezos)

3.3 Blockchain applications:

Blockchain technology, since its inception with Bitcoin, has expanded far beyond just cryptocurrencies. Its fundamental characteristics - decentralization, transparency, immutability, and security - make it suitable for a wide array of applications across various industries. Let's see key areas where blockchain is being implemented:

Decentralized Finance (DeFi)

Decentralized Finance (DeFi) leverages blockchain to recreate traditional financial services in a decentralized manner. This includes platforms for decentralized exchanges (DEXs) enabling peer-to-peer cryptocurrency trading, lending and borrowing protocols that allow users to earn interest or take out loans without intermediaries, and stablecoins designed to maintain a stable value [14]. Yield farming and staking opportunities further allow users to earn rewards on their crypto assets, while decentralized insurance aims to provide transparent and community-driven risk management.

Supply Chain Management

Blockchain technology offers significant improvements to supply chain management by providing immutable and transparent records of a product's journey. This enables enhanced tracking and traceability, crucial for verifying authenticity, monitoring conditions of sensitive goods, and ensuring ethical sourcing [13]. Real-time inventory management becomes possible, leading to greater efficiency and reduced waste, while logistics and shipping processes can be streamlined through automated documentation and payments.

Healthcare

In healthcare, blockchain has the potential to revolutionize data management and security. Securely storing and sharing electronic health records (EHRs) across providers while giving patients more control over their data can improve interoperability and patient outcomes. Furthermore, tracking pharmaceuticals on the blockchain can combat counterfeiting and ensure supply chain integrity, and the technology can enhance transparency and data integrity in clinical trials.

Digital Identity

Blockchain enables the concept of self-sovereign identity (SSI), where individuals control their digital identities and share only necessary information, enhancing privacy and security. This technology can also streamline verification and authentication processes for various services, reducing fraud and improving user experience.

Voting and Governance

Blockchain can enhance the security and transparency of voting systems, potentially increasing trust and participation in elections through tamper-proof and auditable records. Decentralized Autonomous Organizations (DAOs) represent a novel approach to governance, where organizations are managed by code and community proposals, fostering collective decision-making [13].

Real Estate

The real estate industry can benefit from blockchain through the tokenization of assets, allowing for fractional ownership and making property investment more accessible. Additionally, blockchain can streamline property transactions by reducing paperwork and the need for numerous intermediaries.

3.4 Security techniques in blockchains:

3.4.1 Public key cryptography

This mechanism acts as a verifiable proof of who initiated a transaction on the blockchain. It involves a user utilizing a secret, personal key stored in their digital wallet to create a unique mark associated with their transaction data. This mark, along with the transaction, is then shared with the blockchain network for validation [15]. To confirm the transaction's origin, network participants employ a publicly available key that

corresponds to the user's private or secret key. 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 did come from the user.

For example, in Fig. 3, 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. The user then sends its digital signature together with its transaction data to the blockchain network. They perform a cryptographic check to see if the unique mark provided with the transaction was indeed created using the correct secret key associated with the sender. If the check is successful, it cryptographically confirms the transaction's authenticity and that it originated from the owner of the secret key.

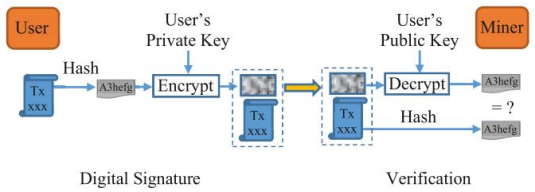

Figure 5. Digital signature and hash used in blockchain transactions [15].

Finally, these digital signatures on transactions and blocks to are used by Ethereum to confirm the identity of the creator and that the signed data has not been modified since signing.

3.4.2 Zero-Knowledge Proofs (ZKPs)

It is a cryptographic technique employed by some blockchains to enhance user privacy. A key application of ZKPs is verifying transaction validity without revealing sensitive information [16]. For instance, when a user initiates a payment, the blockchain needs to confirm sufficient funds. However, with ZKPs, this verification can occur without any need to check the sender's identity or his total balance. In such a scenario, the blockchain gains zero knowledge about the transacting parties or the sender's overall holdings. While not currently supported on Ethereum, the integration of zk-SNARKs, a specific type of ZKP, is part of its future development plans. So, ZKPs ensures privacy for users that their data is not revealed to other users when initiating a transaction. Figure 6 explains the process of interactive ZKPs.

Figure 6. Interactive zero-knowledge proof [9].

Zero-knowledge proofs involve two parties: a prover and a verifier. The prover makes an assertion that his or her proof is valid, which the verifier must approve, without the prover leaking any "knowledge" other than the assertion itself [17]. To understand how this works we need to study the three main properties of zero-knowledge protocols:

1. **Completeness**: A truthful prover can always convince a verifier that a statement is correct.
2. **Soundness**: A false statement cannot be proven true to a verifier, except with a small probability.
3. **Zero-Knowledge**: The verifier does not know any knowledge of the prover's statement other than the truth of the statement.

One of the ZKP systems was discussed earlier by Goldreich, Micali, and Widgerson [18]: "Given a graph G, can you color the nodes with <= 3 colors such that for every edge {u, v} we have f(u) =/= f(v)?"

Firstly, the prover covers all nodes of the graph and assigns color for each node but do not reveal it to the verifier.

Secondly, the verifier randomly chooses an edge of the graph, and the prover reveals the two vertices of the chosen edge. The prover shows that the two vertices are of a different color. If the two vertices are of the same color, we know that the prover is dishonest and does not have the

solution. If the two vertices are of different colors, the verifier has some (but not full) confidence that the prover is telling the truth. We note that the prover has (E-1)/E probability of cheating, where E is the number of edges in the graph. Then we continue to the next step.

Thirdly, the prover begins re-covering the vertices by randomly switching the ordering of the three colors to prevent the prover from simply knowing the colors of a few specific edges. So after each round, the probability of cheating is decreasing and the trust of the verifier that the prover has a valid solution is increasing as follows [17]:

$$P_{cheating} = \left(\frac{E-1}{E}\right)^n \quad (1)$$

3.4.3 zk-SNARKs

Unlike the normal ZKPs, the prover needs to work one time (non-interactive) and the verifier can verify the correctness of the proof without any need to return to the prover. The fundamental scenario involves a prover aiming to convince a verifier that they possess a secret input 'w' which, when used with a public function 'f' and a public input 'x', yields a specific output 'y' (i.e., y = f(w, x)) [18]. The verifier needs to confirm the correctness of 'y' provided by the prover while ensuring no information about the secret 'w' is revealed.

This setup, characterized by completeness, soundness, and zero-knowledge, mirrors the general zero-knowledge proofs discussed previously and underscores the security principles of zk-SNARKs . However, the unique advantages of zk-SNARKs lie in their efficiency, which is achieved through the following crucial requirements:

1. **Non-interactive:** In contrast to the interactive nature of earlier zero-knowledge protocols, like the 3-coloring example, zk-SNARKs require only a single exchange. The prover sends the output 'y' along with a proof string 'p', which is sufficient for the verifier to confirm that 'y' is the correct result, implying the prover's knowledge of the secret 'w'. This one-way communication is a key source of efficiency, as the verifier needs no further queries.
2. **Succinct:** Given a security parameter 'l', the proof 'p' generated by the prover must be of a small, constant size relative to 'l' (denoted as $O_l(1)$. Furthermore, the time required for the verifier to validate this proof should be efficient, expressed as:

O_l (|x| + |y| + |f|) (2)

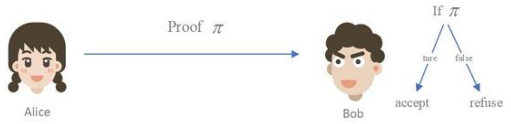

Figure 7. Non-Interactive zero-knowledge proof [9].

3.4.4 Homomorphic Encryption (HE)

Homomorphic Encryption (HE) is a type of encryption that allows computations to be performed on encrypted data **without decrypting it**. The result of those computations, once decrypted, matches the result that would have been obtained had the operations been performed on the original, unencrypted data.

Traditional encryption protects data at rest and during transmission, but once it needs to be processed (e.g., for analytics or computation), it usually has to be **decrypted** – exposing it to potential security risks. So homomorphic encryption is needed to preserve privacy during data processing (especially in cloud computing) and enable secure outsourced computation, such as in medical, financial, or government applications [19]. Suppose you have two sensitive numbers, 5 and 3; you encrypt them using a public key to get Enc(5) and Enc(3), then send them to a server, which, without ever decrypting or knowing the values, performs an operation directly on the encrypted data (like Enc(5) + Enc(3) = Enc(8)), and when you receive the result and decrypt it using your private key, you get Dec(Enc(8)) = 8– meaning the computation was done securely,

privately, and correctly, all while keeping your original data hidden at every step.

At a high level, HE works like this:

1. **Key Generation**:
   A public-private key pair is generated.
2. **Encryption**:
   A user encrypts their data using the **public key**. The data is now unreadable (ciphertext).
3. **Computation on Encrypted Data**:
   A third party (e.g., a cloud server) performs mathematical operations directly on the encrypted data. This party never sees the original data.
4. **Decryption**:
   The result of the computation is still encrypted. The data owner uses their **private key** to decrypt the result, which corresponds to the same result as if the computation was performed on the original data.

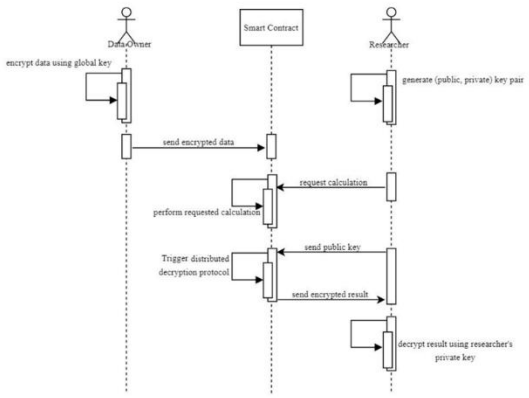

Figure 8. An overview of the proposed system [19].

Table 2. Summary of Cryptographic Methods in Blockchain [9].

Technology	Description	Applications	Advantages	Disadvantages
Public key encryption	Uses public key to encrypt plaintext, and requires private key for decryption.	Encrypted communication, digital signatures, digital certificates.	Secure key exchange, facilitates key exchange.	Slow encryption and decryption, suitable for encrypting small data volumes.
Symmetric key encryption	Uses the same key for encryption and decryption.	Data encryption, communication encryption.	Fast encryption and decryption, suitable for encrypting large data volumes.	Complex key distribution and management, not suitable for scenarios with inconvenient key distribution.
Hash function	Compresses messages of any length into fixed-length hash values for verifying message integrity and consistency.	Data integrity verification, digital signatures.	Efficient, quick verification of message integrity and consistency.	Irreversible, unable to reverse original messages from hash values.
Digital signature	Used to verify the authenticity and integrity of messages.	Data integrity verification, authentication.	Prevents message repudiation, ensures message authenticity.	Low processing efficiency, requires use of public key encryption technology.
Secure multiparty computation	Allows multiple participants to perform computations while keeping data confidential.	Data privacy protection, secure computation.	Protects data privacy, ensures security of multi-party data.	Low processing efficiency, high computational complexity.

Zero-knowledge proof	Proves a statement is correct without revealing the information.	Protects privacy, proves the correctness of a statement.	High computational complexity, difficult to understand and implement.	-

4. Strengths and Limitations of Cryptography in Blockchain

The integration of cryptographic methods into blockchain technology bestows a multitude of significant advantages, contributing to its reputation as a secure and trustworthy system. However, it is equally important to acknowledge the inherent limitations and potential vulnerabilities associated with this reliance on cryptography.

4.1 Strengths of Cryptography in Blockchain

One of the most prominent strengths of cryptography in blockchain is the immutability it provides [9]. Using cryptographic hashing and the chaining mechanism, once a transaction is recorded on the blockchain and becomes part of a block, it is exceptionally difficult to alter or tamper with it [9]. Changing the data in a past block would necessitate recalculating its hash, which would then invalidate the hash of the subsequent block, and so on, effectively breaking the chain and making the alteration immediately apparent to all network participants [9].

Cryptography also enables a high degree of transparency within blockchain networks. While user identities are often represented by pseudonymous public keys, all transactions recorded on a public blockchain are typically visible to anyone with access to the network [8]. This transparency fosters trust and allows for public auditing of the ledger, although it's important to note that the link between a public key and a real-world identity might still be possible to establish in some cases.

One major concern with digital currencies is double spending, which helps to prevent [9]. Digital signatures ensure that a transaction involving a specific digital asset can only be approved by the rightful owner of the private key linked to that asset [9]. This stops bad actors from using the same digital currency fraudulently more than once.

Digital signatures also offer non-repudiation and robust authentication [9]. Recipients can be certain of the sender's identity by using the sender's public key to validate the digital signature of a transaction. The non-repudiation feature makes sure that the sender cannot simply deny starting a transaction after it has been signed and added to the blockchain. Lastly, each blockchain block's data integrity is guaranteed by cryptography, especially through hashing algorithms [9]. Other network users would be able to quickly identify any unauthorized changes to a block's data since they would produce a different hash value for that block [9].

4.2 Limitations of Cryptography in Blockchain

Blockchain's reliance on cryptography has many advantages, but it also has some drawbacks and possible weaknesses. The intricacy of the underlying mathematical issues is intrinsically linked to the security of cryptographic techniques. Even though modern cryptographic algorithms are thought to be resistant to attacks using classical computing, future developments in processing power may make some of these algorithms susceptible [9].

Despite the blockchain's strong cryptographic integrity, attacks can still affect the system. Theoretically, an attacker could manipulate the blockchain by reversing transactions or blocking the confirmation of new transactions in the event of a 51 percent attack, in which one entity or group

controls most of the network's hashing power [8]. However, on big, well-established blockchain networks, such attacks are frequently difficult to carry out and economically prohibitive. Furthermore, even if the underlying blockchain and cryptographic mechanisms are sound, vulnerabilities in smart contracts—self-executing contracts written in code and stored on the blockchain—can be exploited if the code has mistaken or logical flaws [8].

The most important long-term challenge for blockchain cryptography may come from the advent of quantum computers. Quantum computers have the potential to perform certain types of calculations exponentially faster than classical computers, which could have a major impact on the security of currently used asymmetric encryption algorithms, such as RSA and ECC, which are essential for the Bitcoin protocol. Quantum algorithms, such as Shor's algorithm, could theoretically break the security of public key cryptography, potentially allowing for unsecured access to wallets and transactions and compromising digital signatures. Although practical, large-scale quantum computers capable of breaking these algorithms are not yet available, the threat is considered so great that researchers are actively working to develop quantum-based algorithms that can withstand both classical and quantum attacks.

While cryptography is a strong security foundation for the blockchain, it is important to recognize that it is not the only factor determining the overall security of the system. Other aspects such as network consensus mechanisms and the security of Smart Contract Code also play a crucial role. Moreover, responsibility for maintaining the security of digital assets in a distributed ledger system often lies with the individual user, especially as regards the secure management of keys. The potential threat posed by quantum computing requires continued research and development into quantum-proof cryptographic techniques to ensure the long-term security and viability of the blockchains technology in the face of future technological advances.

Conclusion

Blockchain's promise of security, transparency, and decentralization is based on cryptography. Blockchain systems accomplish user authentication, data immutability, and defense against unauthorized tampering and double-spending by combining digital signatures, public-key infrastructure, and cryptographic hashing. The capabilities of blockchain technology to guarantee privacy and secure computation are further expanded by sophisticated cryptographic techniques like homomorphic encryption and zero-knowledge proofs. The complexity of key management, the threat of quantum decryption, and vulnerabilities to 51 percent attacks are some of the significant obstacles the field faces **despite** these advantages. Blockchain systems will need to be future-proofed by implementing improved cryptographic protocols and quantum-resistant algorithms. As blockchain technology develops and spreads into important industries like identity
management, healthcare, and finance, strong and flexible cryptographic frameworks will be essential for preserving resilience, security, and trust.

References

1. Z. Xu, W. Liang, K. C. Li, *et al.*, "A time-sensitive token-based anonymous authentication and dynamic group key agreement scheme for Industry 5.0," *IEEE Trans. Ind. Inform.*, 2021.
2. A. Gorkhali, L. Li, and A. Shrestha, "Blockchain: A literature review," *J. Manag. Anal.*, vol. 7, no. 3, pp. 321-343, 2020.
3. W. Liang, Y. Yang, C. Yang, Y. Hu, S. Xie, K. C. Li, and J. Cao, "PDPChain: A consortium blockchain-based privacy protection scheme for personal data," *IEEE Trans. Rel.*, pp. 1-13, 2022, doi: 10.1109/TR.2022.3190932.
4. H. Nejatollahi, N. Dutt, S. Ray, *et al.*, "Post-quantum lattice-based cryptography implementations: A survey," *ACM Comput. Surv.*, vol. 51, no. 6, pp. 1-41, 2019.

5. N. Denis, M. Laurent, and S. Chabridon, "Integrating usage control into distributed ledger technology for Internet of Things privacy," *IEEE Internet Things J.*, vol. 22, no. 15, pp. 20120-20133, 2023.
6. D. He, R. Wu, X. Li, S. Chan, and M. Guizani, "Detection of vulnerabilities of blockchain smart contracts," *IEEE Internet Things J.*, vol. 10, no. 14, pp. 12178-12185, 2023.
7. X. Luo, X. Chen, X. Chen, and Q. Cheng, "A survey on the application of blockchain in cryptographic protocols," *Cybersecurity*, vol. 7, no. 79, 2024, doi: 10.1186/s42400-024-00324-7.
8. F. M. Saeidia, M. H. Zahedi, and E. Farahani, "A secure and reliable model for financial documents using digital signature and blockchain technology," *AI Tech. Behav. Soc. Sci.*, vol. 3, no. 1, pp. 23-33, 2025.
9. S. W. Lo, Y. Wang, and D. K. C. Lee, "Cryptography and blockchain technology," in *Foundations for Fintech*, Singapore: World Scientific, 2021, pp. 1-30.
10. National Institute of Standards and Technology, "Blockchain technology overview," *NIST Interagency Report 8202*, 2018.
11. S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf
12. S. Lin, "Proof of work vs. proof of stake in cryptocurrency," 2023.
13. Ugwu, J. N., Hassan, J. B., Akinsanya, S. E., Ayogu, B. A., Adubi, S. S., & Olagunju, M., "Comparative Analysis of Eight Different Blockchain Technology Schemes and Their Implementations," University of Ibadan Journal of Science and Logics in ICT Research (UIJSLICTR), 2024.
14. Connors, C., & Sarkar, D. (2022). Comparative Study of Blockchain Development Platforms: Features and Applications. arXiv preprint arXiv:2210.01913.
15. B. Sriman, S. G. Kumar, and Shamili, "Blockchain technology: Consensus protocol proof of work and proof of stake," 2020.
16. H. Guo and X. Yu, "A survey on blockchain technology and its security," *Blockchain: Res. Appl.*, vol. 3, no. 2, p. 100067, 2022.
17. O. Goldreich, "A short tutorial of zero-knowledge," Weizmann Institute of Science, 2010.
18. T. Jo, "Bitcoin and the proof-of-work consensus mechanism: Energy consumption and scalability challenges," 2020.
19. M. M. Ali, M. S. Alajlan, and S. H. Almotiri, "Blockchain and homomorphic encryption for data security and statistical privacy," *Electronics*, 2020.