International Journal of Safety and Security Engineering
Vol. 16, No. 3, March, 2026, pp.701-709
Journal homepage: http://iieta.org/journals/ijsse

# Security Threats, Performance Trade-Offs, and Mitigation Strategies in Blockchain-Based Educational Credential Systems: A Comprehensive Review

Pooja M. Pondkule[1,2*], Sonali Kothari[1]

[1] Department of Computer Science and Engineering, Symbiosis Institute of Technology, Symbiosis International (Deemed University), Pune 412115, India
[2] Department of Computer Science and Engineering, Dnyanshree Institute of Engineering and Technology, Satara 415001, India

Corresponding Author Email: puja.pondkule.phd2022@sitpune.edu.in



https://doi.org/10.18280/ijsse.160320

**Received:** 27 January 2026
**Revised:** 12 March 2026
**Accepted:** 20 March 2026
**Available online:** 31 March 2026

***Keywords:***
*blockchain, educational credential systems, security threats, performance trade-offs, mitigation strategies, smart contracts, decentralized systems*

**ABSTRACT**

The blockchain technology has proven itself in the education industry that it has the capability to improve the security, validity and transparency of academic qualifications. The current research paper discusses the use of blockchain-based solution to issue and confirm educational certificates. It provides a detailed explanation of the most significant security weaknesses and threats such as network-level attacks and smart contract threats, and some of the mitigation measures that have been suggested to increase the stability of the system. The study also covers key performance factors, which determine the viability of the application of the blockchain technologies in the educational environment, including the scalability, latency, throughput, privacy, interoperability and cost-effectiveness. The article indicates the possibility of blockchain providing verifiable and tamper-resistant academic records and addressing the issues with trust and data integrity, relying on the research of the existing literature.

## 1. INTRODUCTION

Blockchain is a decentralized and Distributed Ledger Technology (DLT) that makes a recording of transactions with a number of computer nodes in a clear, safe, and permanent way [1]. All transactions are confirmed into a block and pack successively together to form a chain making a permanent record of transactions. Starting in cryptocurrency, blockchain has progressed side by side with the rise of smart contracts and shows massive potential across a wide range of industries by helping to implement more robust government systems and address the growing threat of ecological and financial instability [2]. The possibility of the risks posed by the forged certificates to the safety of the masses (including the forging of professional qualification) underscores the importance of strong risk assessment systems, and effective crisis management and disaster response plans, especially in those instances where fake credentials can result in critical malfunctions in medical, engineering, or political processes. Recent technological progress has led to more people being interested in the use of blockchain in the management of educational records and credentials. The intrinsic properties of decentralization, transparency, and immutability make blockchain a promising solution to address the limitations of conventional educational records keeping systems, especially removing single points of controls by distributing the data, thus making blockchain more secure against attacks or against the loss of data through accidental destruction [3, 4]. Also, blockchain is impossible to change, so after data is put into a block, no one can modify it, thus education records remain intact. There are however challenges to the adoption of blockchain-based educational credential systems. The risk of security threats arising on the blockchain foundation itself, such as allowing unauthorized access or manipulating data using malicious attacks, is one of the threats. Also, the sophistication of blockchain can be a disadvantage to the general implementation since learning institutions might have a hard time learning how to make effective use of the technology. The other important issue is the performance parameters of blockchain-based educational credential systems [5]. Though the blockchain technique brings with itself the advantages of increased security and visibility, this might be an expense towards slower processing and accessibility by a lesser number of transactions, which will affect user experience and accessibility of the overall system [6]. These Security concerns and performance limits are even more acute in the field of issuing and authenticating education qualifications where integrity, scalability and reliability are paramount.

Specifically, the review is a specific analysis of blockchain based credential systems in education, and in particular, certificate issuance and verification [7, 8]. The research questions of the study are as follows:

(1) Which are the main security threats and vulnerabilities of these systems?

(2) What is the performance parameter that has an impact on their practical deployment and scalability?

(3) What mitigation implies have been proposed in order to curb these problems?

This paper will be limited to literature based research of

security and performance variables which shall provide a systematic understanding on the interaction between the two parameters in reference to the educational uses [9-11].

### 1.1 Overview of blockchain

Blockchain is a novel technology, which is a distributed and decentralized digital registry. A conventional ledger is controlled by a single entity that is not powerful in the hands of one or the other user. Rather, it is supported through a collection of computers. Blocking of transactions takes place. The blocks include a group of authenticated transactions, a time stamp, and a cryptographic connection to the last block. These blocks are connected in a chronological order as well as through cryptography into a chain. Such chain renders it computationally infeasible to alter under the assumption of honest majority past records since changing a single block would necessitate changing all the other blocks. The data of a blockchain is not stored on a central node (such as a bank), but it is shared by many computers (nodes) on a peer-to-peer network. This increases its resilience to single points of failure and censorship. After a block has been appended to the chain it is very hard, almost impossible, to change or remove the information unless the whole network agrees to it. This guarantees integrity and permanence of records. Most blockchain systems (particularly those in the public) are fully transparent to everyone, though the names of the participants may be fake names. This openness creates trust amongst people. To ensure that all participants agree on the validity of transactions and the state of the blockchain, various "consensus mechanisms" are employed [12]. Common examples include:

Proof-of-Work (PoW): Network participants (miners) solve complex computational puzzles to validate transactions and create new blocks [13]. This is used by Bitcoin.

Proof-of-Stake (PoS): Validators are chosen to create new blocks based on the amount of cryptocurrency they "stake" or lock up. This is more energy-efficient than PoW and is used by Ethereum (after its Merge) [14].

### 1.2 Layered architecture of blockchain

Layered architecture of blockchain as shown in Figure 1. offers an organized way of figuring out its diverse parts as well as the relationships between these parts. The architecture consists of five basic layers.

**Hardware Layer:** This is the lowest layer and it is parts that make up the physical infrastructure that supports the blockchian network. It consists of the set of computers (nodes), servers and other hardware that the blockchain needs to be able to work. This layer simply comprises the universality of all the stakeholders in the blockchain network. The devices or nodes, contribute computation power, and storage independently to the network. Mining hardware is also part of this layer in PoW systems such as the Bitcoin system. The connectivity of these devices is used as the base of data sharing and processing of transactions.

**Data Layer:** The layer deals with the packaging and storing of all transaction data in the block chain. It establishes the design of data structures and ways in which they are stored and protected. Blocks are the basic components of a blockchain that are constituted by transaction bundles. The blocks are arranged in an unchangeable chain, and each of them consists of transaction information, timestamps, and hash of the preceding block. Merkle trees are normally used in the data layer as well and allow efficient summarization and verification of the integrity of the transactions in a block. This layer underlines tamper-proof and transparent book-keeping that is the main idea of blockchain technology.

**Network Layer:** The network layer controls the manner in which nodes within the blockchain network initiate communications amongst one another. It defines the protocols of transaction and block diffusion in the network. Normally using peer-to-peer (P2P) protocols, this layer is what most participating nodes can find each other, share the information about transactions and remain synchronized to the actual state of the block chain. It controls the creation of blocks and the addition of blocks, as well as node discovery and supports the decentralization of the network.

**Consensus Layer:** This important layer makes sure that parties in the network reach a consensus regarding the validity of transactions as well as the sequence of the blocks that are added to the blockchain. It builds trust in distributed setting. Some of the consensus mechanisms that are carried out by a consensus layer are PoW, PoS and so on. These mechanisms stipulate how nodes can verify the transactions, suggest new blocks, and they must come to a group consensus regarding the state of the ledger. This stratum is critical towards avoiding cases of double-spending and safeguarding the integrity and safety of the blockchain.

**Application Layer:** This is the layer where the users work with on the direct level. It includes the applications, smart contracts and user interfaces that rest on the blockchain infrastructure. The layer will use services of the lower layers to provide support to different use cases. This layer is home to so-called Decentralized Applications (dApps), smart contracts (self-executing agreements written in code), digital wallets, and others. It outlines the way the base technology of blockchain can address practical issues, such as Decentralized Finance (DeFi) and Non-Fungible Tokens (NFT), supply chain and voting systems.

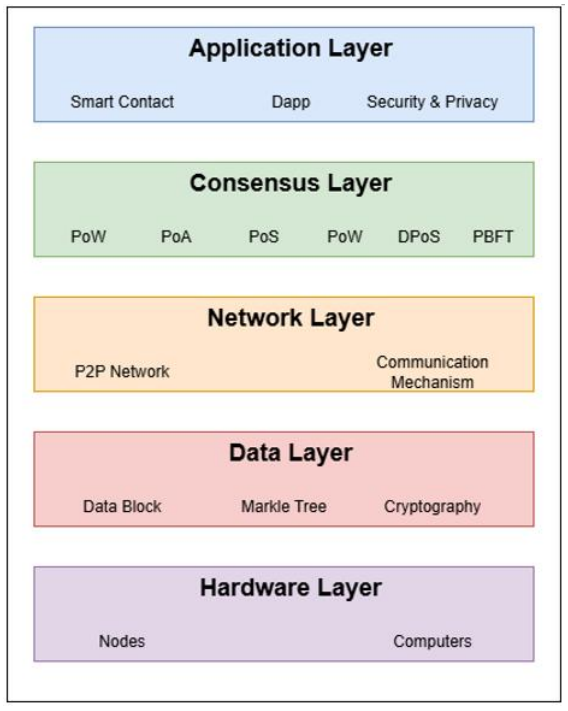


**Figure 1.** Layered architecture of blockchain

## 2. LITERATURE REVIEW

### 2.1 Attacks on various layers of blockchain

As illustrated in Figure 2, various types of attacks can occur across different layers of the blockchain architecture.

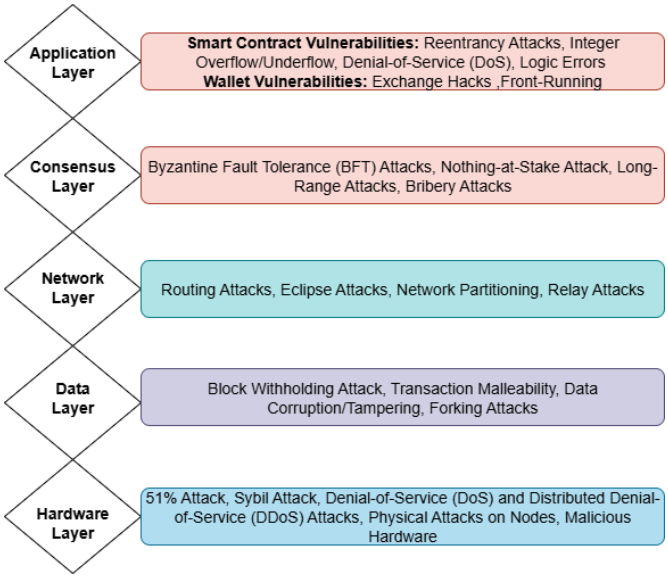


**Figure 2.** Attacks on various layers of blockchain

2.1.1 Application layer attacks

**(1) Smart Contract Vulnerabilities**

Smart contract vulnerabilities originated as a result of contract code flaws and could be used to disrupt the operations of blockchains.

A reentrancy attack is a type of attack in which a malicious contract repeatedly invokes another contract before the original execution has been done, which can cause the state to be changed or the resource to be exhausted. In educational credential systems, such vulnerabilities could allow unauthorized modification or duplication of certificate records.

Reentrancy Analyzer (RA) that integrates symbolic execution and equivalence verification with a satisfiability modulo theories solver in order to examine Ethereum Virtual Machine (EVM) interactions at the level of the EVM bytecode language. Other prevention strategies are updating the user balances prior to making external calls, limiting the unreliable functions, and using mutex lock to avoid the occurrence of repetitive functions [15]. These measures counter the critical effects of reentrancy vulnerabilities of smart contracts.

Integer overflow happens when an arithmetic operation has surpassed the maximum value that it can represent, whereas integer underflow happens when the operation has an outcome that is below the minimum possible. In smart contracts with the data type being uint 256, overflow may happen where the values may surpass $2^{256}$-1 and underflow may happen where the values may fall below zero, yielding wrong computations. These vulnerabilities may undermine the logic of contracts, especially in balancing. To illustrate, an underflow can cause a balance to be reset to a big number so that it becomes possible to perform unauthorized transactions [16]. The most notable examples are the 2018 Beauty Chain attack, in which an overflow could be exploited to withdraw more tokens than usual, and the Proof of Weak Hands exploit, in which the underflow could steal 866 ETH. These cases highlight the risks of inadequate arithmetic checks in smart contracts.

Use solidity compiler versions 8.0 and higher to prevent these problems. As well, the problem of underflow and overflow can be prevented when the code of the smart contract is tested and audited before being deployed.

Denial-of-Service (DoS) attack on smart contracts is the condition when the attacker uses some vulnerabilities to disrupt the normal functioning of the contracts, making them inaccessible to their legal users [17]. In comparison with classic DoS attacks, which saturate servers, smart contract DoS attacks may affect excessive gas usage, logic errors, or locking of contract state, and no additional interaction should take place. Indicatively, recursive functions executed in loops over dynamic data structures may be used to create memory-space exhaustions, resulting in an outright crash. Likewise, operations that are intended to fail (e.g., absence of fallback() or receive() functions) can be blocking. These attacks can block money, limit legitimate transactions, and interfere with dApps. These weaknesses may be extremely harmful in case the developers do not solve them in the initial stages of development because of the immutability of smart contracts.

The mitigation measures are not to use unbounded loops and large dynamic data structures, but rather to perform batch processing or off-chain computing. Access control systems are used to restrain important operations, and the use of secure code patterns like the checks-effects-interactions pattern will minimize the risks associated with external calls. Other controls are circuit breakers, rate limiting, sound estimation of gas, input validation, and comprehensive testing with edge cases to establish weaknesses prior to deployment.

Smart contracts contain logic errors that cause unexpected or malicious behavior as a result of the implementation of business logic errors. Such problems are typically due to improper management of edge cases, misconceptions about how other contracts interact or do not update their contract states correctly. As an example, an error in logic can provide access to repeated withdrawals because of incorrect updating of the balance. As compared to low-level vulnerabilities, logic errors are more difficult to spot since the binaries run properly but result in unintentional actions. These imperfections may result in monetary losses, unauthorized access, or the loss of a contract, which is very dangerous with dApps.

Mitigation involves specification of contract logic clearly, extensive testing and formal verification of the contract logic. Early failure detection techniques include unit testing, fuzzing and Test-Driven Development (TDD). Require and assert functions are used to verify input to ensure proper state transitions, and secure design referents such as checks-effects-interactions minimize risks of external interactions. Also, formal verification (e.g., Certora, MythX, Slither) and security audit by an independent third party are essential to identify the subtle logic errors prior to deployment.

**(2) Wallet Vulnerabilities**

The vulnerabilities of the wallets are based on the vulnerabilities in software or hardware wallets that are used to store and handle the private keys and other digital assets. They can be either software (e.g. browser extensions or mobile apps) or hardware wallets, and be used either by malware injection, phishing, Man-In-The-Middle (MITM) attack, extraction of keys, or by compromising the supply chain. As an example, user credentials can be stolen by malicious browser extensions or by fraudulent interfaces on wallets, whereas hardware wallets could be exposed to physical access or via security bugs in firmware. Blockchain transactions are permanent, and in the case of lost private keys, one could lose all their assets.

Some of the mitigation measures include code signing, regular update, and sandboxed environments as far as the design of a secure wallet is concerned. Security is improved by using Multi-Factor Authentication (MFA), the integration of a hardware wallet, and transaction confirmation mechanisms. In the case of hardware wallets, the risks can be mitigated through updating the firmware and buying them in reputable platforms. It is also enhanced by the use of non-custodial and open-source wallets that enhance transparency and control. Moreover, the awareness of users about backup procedures and the safety of seed phrases is necessary to avoid wallet-related.

Exchange hacks are dedicated to cryptocurrency trading sites and are based on poor key management, lack of access control, software vulnerabilities, or insider threats. Attackers commonly seek to steal personal keys, account funds or sensitive account information. The high financial consequences of the attacks are indicated by the notable incidents, such as the Mt. Gox (2014) and Coincheck (2018) breaches. Centralized exchanges are most susceptible as they are one of the single points of failure [18].

The mitigation measures comprise saving the majority of funds in offline cold wallets, multi- signature (multisig) schemes, routine security audits, penetration tests, and bug bounties. Two-Factor Authentication (2FA), IP whitelisting, and anti-phishing measures are strong authentication tools that add to the security. Moreover, the possible losses can be minimized due to regulatory compliance and insurance systems. The users should only use trusted exchanges and do not leave huge amounts of money stored on centralized platforms (not your keys, not your coins) [19].

A front-running attack is an attack in which a malicious user notices a pending operation in the mempool and sends a competing operation at a higher fee in order to have priority of execution. This takes advantage of transaction ordering protocols in transparent blockchain solutions. Usually, with DeFi and NFT markets, front-running is a method of attackers to make a profit due to manipulation of transaction timing, providing unfair advantages, price manipulation, and creating less trust in users [20, 21].

The mitigation measures are aimed at decreasing the visibility of transactions and the manipulation of orders. Methods involve commit-reveal schemes to hide details of transaction, private transaction relays (e.g., Flashbots) to avoid public memorools as well as slippage limits to make such attacks less profitable. Furthermore, randomization of the sequence of network transactions, the use of fair transactions and security audits of Miner Extractable Value (MEVs) are aimed at enhancing the resilience of the system [22].

2.1.2 Consensus layer attacks

(1) Byzantine Fault Tolerance (BFT) attacks are an attack on the consensus mechanism of distributed blockchain systems, as well as faulty nodes or malicious nodes that can act dishonestly, collude, or relay conflicting messages. This may cause honest nodes to fail to agree on a consensus, resulting in a variety of problems: in permissioned systems that have less validators, including double-spending, chain forks and network failure. Overall, systems based on BFT are vulnerable in case more than a third of the nodes are compromised [23, 24].

Mitigation is achieved through strong consensus protocols like Practical Byzantine Fault Tolerance (PBFT) and variants thereof, capable of surviving a small number of bad nodes. Other techniques are: validator rotation, randomized leader choice, reputation system, secure protocol communication, and formal verification of the protocol to improve resilience [23, 24].

(2) Nothing-at-Stake attack is a problem that can take place in PoS systems, where validators can serve in two or more competing chains because the cost of computing is low, which increases the chances of forks, double-spending, and network instability [25, 26].

Reducing the conditions that punish fraudulent validators, finality like Ethereum Casper FFG, random selection of validators, and delayed incentive rewarding are some of the mitigation strategies. These strategies provide economic reasons against malice actions and ensure consensus integrity [26].

(3) Long-range attacks have an impact on PoS blockchains because attackers may utilize the past valid minor keys to generate an alternative past chain. It can cause misleading new or offline nodes and weaken chain finality, which may cause double-spending or network partitioning [27].

Countermeasures such as finality rules, which ensure that previously confirmed blocks cannot be rolled back, checkpointing, weak subjectivity assumptions with trusted snapshots, and key rotation are some of the notable countermeasures. Such mechanisms are implemented in protocols like Ethereum (Casper FFG) and Polkadot (GRANDPA) to provide integrity in the chain [27].

(4) A bribery attack happens when a malicious actor pays off the validators or miners to break the protocol rules, including transaction reordering, activity or malicious forks. Such attacks are easier in PoS and delegated consensus systems, where it is frequently less resource-demanding to influence the validators under these systems compared to PoW systems. This type of manipulation may allow for spending twice or make biasing governance choices, which is a threat to the integrity of the network [28].

The mitigation tactics involve the reduction of mechanisms to punish untruthful action, higher staking, and delayed rewards to deter the short-term incentive. Randomized validator selection and confidential ballot eliminate the chances of colluding specifically. Also, there are governance mechanisms and peer observation that led to the detection and prevention of bribery attempts and maintain equity in consensus [28].

2.1.3 Network layer attacks

(1) The routing attacks (e.g., BGP Hijacking) use the vulnerabilities of the internet routing protocols, including BGP to intercept, delay, or modify blockchain traffic. The attackers may divide the network, interrupt communications, or provide incoherent blockchain views, which may allow them to double-spend or stall consensus. As an example, a BGP hijack will be able to isolate a set of nodes to feed them with wrong or delayed data [29].

Encrypted communication (e.g., TLS), redundant network paths, and the selection of peers across both geographic and network boundaries are all mitigation measures. The analysis of BGP abnormalities and the implementation of safety improvements, such as RPKI mitigate the risks as well. The communication resilience is also enhanced by overlay and relay networks (e.g., FIBRE).

(2) An eclipse attack takes out a node by dominating both its in and out connections in such a way that attackers are able to disrupt its knowledge of the blockchain. This may result in

twofold spending, rapacious mining or agreement manipulation. They may be in such a form of attacks that tend to utilize a bad peer selection in addition to flooding the nodes with bad IP addresses [30].

Randomized and diverse peer countermeasures were selected, such as blocking associations in the same IP range, using peer rotation, and trusted nodes. Other technologies as well as DNS seeding and whitelisting, will reduce the probability of node isolation [31, 32].

(3) The blockchain network is partitioned into network partitioning broken components, and node-to-node communication is blocked. This can create blockchain inconsistencies, forks, and two-spends attacks. BGP can occasion partitions hijacking, DNS attacks, or DDoS attacks.

Mitigation involves the economic implementation of geographically redundant connections, the distributed nodes, and the implementation of secure routing processes. Resilience of Relay networks enhance the partitioning attacks, encrypted communication, anomaly detection systems, and encrypted communication.

(4) Relay attacks occur in a situation where a hacker gets into and sends the messages between two parties again without their consciousness which makes them believe that they are communicating directly. This kind of attack is common when it comes to wallet transactions authentication protocols particularly in NFC or Bluetooth and may lead to attacks of unauthorized transactions and does not touch any personal key.

The mitigation techniques include mutual to prevent replay attacks nonces and timestamps to prevent replay attacks authentication, proximity verification and secure session control. Reduction of the threat of attack via relay still further is accomplished by making use of end-to-end encryption and other means authentication with biometrics authentication.

2.1.4. Data layer attacks

(1) An insider attack is a block withholding attack in PoW mining pools that a malicious miner does not provide valid blocks, and in its place sends partial proofs (shares) to one to seem valid. This reduces the overall rewards of the pool and can be used in undercutting competing pools, and results in an ultimate effect on trust and decentralization [33].

Mitigating is founded on statistical anomalies detection mining behavior, rewarding programs, Pay-Per-Last-NShare (PPLNS) and cryptographic evidence of truthful mining. These methods detract from the withholding and enhance equity in mining pools.

(2) Transaction malleability enables the attacker to alter the identifier (hash) of a transaction, but not its contents, which can interfere with the operation of other transactions (or also allow the attacker to spend the same money twice). In the Mt. Gox case, this weakness was taken advantage of (Decker and Wattenhofer) [34].

The Segregated Witness (SegWit) update alleviates this problem by splitting signature information into transaction hashes. Other initiatives involve recording transactions in the form of outputs rather than identification and the use of standardized transaction formats to promote integrity.

(3) Data corruption happens when someone makes changes to data before it is recorded on the blockchain or while it is being sent. The blockchain is very good at keeping data safe once it is confirmed. There are some weaknesses in the way data is put into the blockchain in smart contracts or with bad nodes that can cause problems with the data. To stop this from happening, we can use codes called cryptographic hashes, digital signatures and Merkle proofs. We also need to make sure the data we put in is good and valid. The data can also be kept authentic and reliable with the help of the so-called decentralized oracles such as Chainlink, and secure methods of reaching an agreement on data.

(4) Forking attacks refer to a situation as an individual creates another branch of the blockchain to alter transactions or use money twice. It is a frequent occurrence in systems consuming a large quantity of computer power, whereby the longest blockchain is victorious. In case a person possesses more computer power, he or she can cheat the system so that it picks his or her chain in the manner of Eyal and Sirer of 2014.

To discourage we may employ mechanisms that ensure that transactions are final and irreversible. Individuals can also be rewarded to be honest. Punish them in case they attempt to produce false blocks. Proofs of stake Systems which employ a mechanism of securing the blockchain are also safer as they possess regulations that deter malicious acts and make the creation of a counterfeit chain more difficult to an individual.

**Hardware/Infrastructure Layer Attacks:**

(1) A 51% attack (majority attack) occurs when an entity gains majority control of a network's computational power (PoW) or stake (PoS), enabling manipulation of the blockchain. This allows double-spending, transaction censorship, and chain reorganization by overriding the honest chain. Such attacks undermine decentralization and trust, particularly in smaller networks with lower hash power, as seen in Ethereum Classic and Bitcoin Gold [35]. This could enable attackers to alter or invalidate issued academic credentials.

Mitigation focuses on increasing decentralization of mining or staking power. Systems like penalties reduction in PoS, block finality checkpointing and hybrid consensus mechanisms (e.g., PoW + PoS) make attacks less possible. Network surveillance and dynamically adjusted difficulty also assistance in the monitoring and reaction to abnormal conduct.

(2) Sybil attack represents a type of attack where a number of fake identities are formed to have disproportionate power in a blockchain network. This may interfere with consensus, corrupt governance or isolate honest nodes. Uncontrolled node creation can also be quite detrimental to permissionless systems [36]. Decentralized academic networks may be manipulated by fake identities.

Sybil-resistant consensus mechanisms like PoW and PoS are considered as the defense mechanisms, in which influence is potentially derived based on resources or stake. Others are reputation systems, identity checks, implementation of peer diversity and evidence-of-personhood. Permissioned blockchains also minimize the risk associated with blockchain as only trusted parties can participate.

(3) DoS/DDoS attacks will interfere with the availability of blockchains by flooding nodes, smart contracts, or network resources with excessive requests. Such attacks may either slow down transactions, block propagation or put nodes offline. A notable example is the Ethereum DoS attack of 2016, which took advantage of inefficient operations to waste a lot of resources. This can postpone or interrupt the certification verification provision to institutions and employers.

The strategies that can be employed to mitigate these issues are rate limiting, resource pricing schemes (e.g., gas fees), and request validation to make the abuse less appealing. Other protection features are DDoS protection services, redundancy of nodes and adaptive filtering. The protocols are also improved, like the EIP-150 in Ethereum, making it more resilient to resource-exhaustion attacks.

(4) Physical attacks are directed to blockchain nodes and focus on hardware access, i.e., full nodes, validator machines, or wallet devices. Attackers can steal private keys, cause havoc, or destroy hardware using such methods as hardware probing or side-channel analysis. Such threats are greater in unsecured setups, e.g., personal devices or insecure infrastructure. With PoS, a breached node can be used to sign blocks or even misuse staked funds.

Strong physical security measures that are part of mitigation are tamper-evident hardware, secure boots, and Hardware Security Modules (HSMs) to protect keys. Other security measures are biometric authentication and PIN protection, geo-redundancy, backups and cold/air-gapped storage of critical keys, minimizing the effects of physical compromise.

(5) In Malicious hardware attacks the device is compromised or there is a backdoor to steal sensitive data or interfere with the work of the blockchain. Hardware wallets, ASIC miners, and validator nodes are the threats that are especially applicable procured out of unreliable sources. As an example, an interfered device can leak private keys or permit unauthorized transactions without being noticed with secretive backdoors to the firmware.

Some of the risk mitigation measures are going to source the devices from reputable vendors, transparency with open-source firmware, and so on updating of firmware regularly. Other solutions, like cryptographic attestation, tampering, and air-gapped architectures, increase security. Trust in hardware components is also enhanced by certification standards (e.g., FIPS 140-2, Common Criteria), and third-party audits.

An overview of the attacks in various layers reveals that educational credential systems are affected the most by the application-layer vulnerabilities, such as smart contract and wallet attacks. These weaknesses may compromise the integrity of the certificate and the ownership of its users. Network-layer attacks, including routing and eclipse attacks, on the other hand, mostly affect the availability and communication reliability. Other systemic attacks, such as BFT attacks and 51% are less prevalent in well-established networks but have consensus-layer attacks such as the 51% and BFT attacks. Furthermore, priorities of mitigation are different. The problems of the application layer require secure coding and auditing, whereas the network and consensus threats rely more on the design of the protocols and infrastructure protection.

## 3. RESULTS AND DISCUSSION

The relevant performance parameters, which will be used in this research, comprise security considerations with key performance parameters applicable in blockchain-based educational credential systems (Table 1). The findings indicate that although blockchain technology has a great potential in the field of integrity, transparency, and irreversibility of academic credentials (functionalities that resolve issues related to the traditional record-keeping system such as forgery, data tampering, and unverifiability), these advantages are inextricably linked to the system-level limitations, especially in terms of performance and resource demands.

**Table 1.** Analysis of performance parameters in blockchain-based educational credentialing [15, 16, 18, 37-43]

| Performance Parameter | Description | Relevance to Education System | Challenges | Potential Solutions / Tools |
|---|---|---|---|---|
| Scalability | Ability to handle growing number of users/ transactions | Supports mass issuance and verification of certificates by institutions | High volume may slow public blockchains | Use permissioned blockchains (e.g., Hyperledger), Layer-2 scaling |
| Latency | Time delay in transaction confirmation | Needed for real-time certificate verification by employers | Delays in PoW systems | Use PBFT or PoA consensus; optimize network latency |
| Security | Protection from fraud, tampering, unauthorized access | Prevents certificate forgery and ensures authenticity | Vulnerabilities in smart contracts or network layer | Use digital signatures, secure hashing, formal verification |
| Immutability | Once written, data cannot be changed | Ensures certificates cannot be altered after issuance | Conflict between immutability and error correction | Use off-chain revocation lists; issue new entries with updates |
| Privacy Compliance | Alignment with data protection regulations (e.g., GDPR) | Sensitive student data must remain private | Storing personal data on-chain violates privacy laws | Store only hashes on-chain; use off-chain encrypted storage |
| Interoperability | Compatibility with external systems/databases | Allows global recognition of degrees; easy integration with employer systems | Varying data formats and blockchain protocols | Use open standards (e.g., Blockcerts); API integrations |
| Throughput | Number of transactions the system can handle per second | Required during exam results or graduation seasons | Low throughput causes bottlenecks | Use high-throughput blockchains (e.g., Solana, Hyperledger Fabric) |
| Auditability | Ability to track history of data entries and accesses | Builds trust and supports dispute resolution | Maintaining audit trails while preserving privacy | Use cryptographic logs; transparent ledgers with access control |
| Usability & Accessibility | Ease of use by students, institutions, verifiers | Ensures wide adoption and minimal user errors | Complex interfaces, tech literacy barriers | Mobile apps, multilingual UI, LMS integration |

### 3.1 Synthesis of findings

Besides, the security threats discussed above demonstrate that the vulnerabilities in different layers (application, network, and consensus) of the system, including smart contract vulnerabilities that impair credential integrity and network-level attacks that slack the verification process, can directly influence system performance and reliability, which is why a

design approach that takes security and performance into the proper balance can be considered.

### 3.2 Trade-off analysis

This paper also demonstrates that there are trade-offs inherent between the performance parameters: A secure system may be faster, safer, and less expensive by using advanced cryptography and strong consensus mechanisms; a highly secure consensus mechanism may in turn take up unrealistically long time to process transactions and thus negatively affect the real-time verification of academic credentials. Similarly, to ensure the privacy of sensitive student information the required privacy preserving mechanisms may impose even further computational overhead and decrease the efficiency of the system at the expense of scalability and speed optimization which is the case in permissioned blockchain systems, can be used to improve performance at the cost of reduced decentralization and increased reliance on trusted entities. These tradeoffs suggest that there is no single blockchain setup that can maximize all parameters simultaneously, and thus system designers will need to be careful to trade off these parameters according to the desired use and scale of usage.

### 3.3 Educational credential system

With education credentialing, a few of these parameters are more significant than others; security and data integrity, e.g. take priority since a breach of either may directly affect the integrity of the academic record, and privacy compliance is also mandatory, as the educational records and information of students are sensitive. Such other performance factors as scalability and latency are relevant, but can be tuned to the requirements of the institution and anticipated outload on the system (e.g., credential verification systems may well need to be highly reliable and accurate, it need not be having a high transaction throughput). The other parameter is interoperability because the educational systems are usually required to interface with various institutions, employers, as well as verification platforms. A secondary parameter is cost-effectiveness, especially to large scale deployments. infrastructure and operation cost is a factor, but eventually, the scalability, latency, reliability and accuracy and interoperability should take center stage, depending on the special needs of educational ecosystem.

### 3.4 Practical deployment considerations

Although the optimal design to use blockchain in educational records does not exist, the trade-offs will vary depending on the context, which includes scale, trust model, and data sensitivity. Unlike public blockchains which are typically more appropriate in a cross-border credential verification which requires transparency and global trust, permissioned blockchains are more appropriate and institutional in use because of improved scalability, governance, and regulatory compliance. The storage that is immutable is not scalable to large amounts of academic records, thus it is preferable to rely on off-chain storage with on-chain verification. These choices also have trade-offs between decentralization and performance, privacy and auditability, and will use APIs to be able to integrate with the current institutional systems but are associated with interface-level security concerns. Therefore, application specific needs like scale, trust model and data sensitivity have to be taken into account in deployment decisions.

To address governance and transparency of process of educational credential systems, the framework suggested as shown in Figure 3. comprises of an operational model where issuing authorities are educational institutions, students are credential holders, verifiers are employers or academic institutions, and permissioned blockchain network where accredited educational institutions are the issuing authorities whose identities are cryptographically verified and authorized to issue, update, and revoke credentials using smart contract functionality, students are credential holders, verifiers are validators of credentials by family querying the blockchain ledger to verify the issuer, the data, and the revocation status without intermediate. The institutional nodes operate the permissioned blockchain network and guarantee governance, controlled participation and transparency.

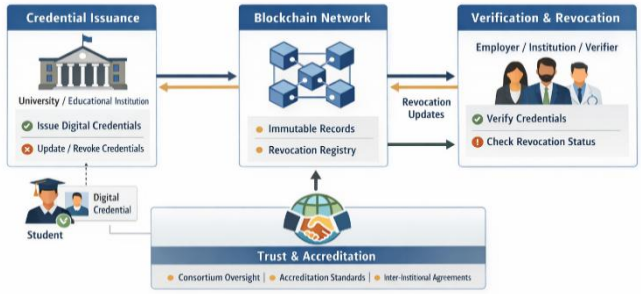


**Figure 3.** Proposed model

## 4. CONCLUSION

The paper is an in-depth discussion of security threats, vulnerabilities, performance parameters and mitigation measures in blockchain-based educational credential systems that reveal that blockchain technology has the potential to offer significant gains to ensure the validity, traceability, and permanence of academic records without addressing some of the most significant flaws of traditional record-keeping systems. The examination of attack vectors at the application, network, consensus, and system layers reveals that security remains a major issue, with issues around smart contract vulnerabilities, wallet security, and consensus manipulation. In contrast, the assessment of performance parameters indicates that issues related to scalability, latency, privacy, interoperability, and cost have a major impact on the practical implementation of blockchain solutions in educational settings. This work contributes to an understanding of the inherent trade-offs between security and performance parameters in blockchain-based educational credential systems. In general, these results indicate that the adoption of blockchain in education is a context-sensitive and not a one-size-fits-all solution, and that future research should be aimed at designing optimized and hybrid frameworks that balance these trade-offs more effectively, as well as at examining real-world implementations and standardization initiatives to enhance interoperability and large-scale adoption.

## REFERENCES

[1] Pondkule, P.M., Kothari, S. (2025). Implementation of

blockchain-based document management system for higher education organizations. International Journal on Smart Sensing and Intelligent Systems, 18(1): 1-11. https://doi.org/10.2478/ijssis-2025-0001

[2] Luu, L., Chu, D.H., Olickel, H., Saxena, P., Hobor, A. (2016). Making smart contracts smarter. In CCS '16: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, pp. 254-269. https://doi.org/10.1145/2976749.2978309

[3] Atzei, N., Bartoletti, M., Cimoli, T. (2017). A survey of attacks on Ethereum smart contracts (Sok). In International Conference on Principles of Security and Trust, Springer Berlin Heidelberg, pp. 164-186. https://doi.org/10.1007/978-3-662-54455-6_8

[4] ConsenSys Diligence. (2021). Smart contract security best practices. https://consensys.github.io/smart-contract-best-practices/.

[5] Eskandari, S., Clark, J., Barrera, D., Stobert, E. (2018). A first look at the usability of bitcoin key management. arXiv preprint arXiv:1802.04351. https://doi.org/10.48550/arXiv.1802.04351

[6] Chu, H., Zhang, P., Dong, H., Xiao, Y., Ji, S., Li, W. (2023). A survey on smart contract vulnerabilities: Data sources, detection and repair. Information and Software Technology, 159: 107221. https://doi.org/10.1016/j.infsof.2023.107221

[7] Chen, G., Xu, B., Lu, M., Chen, N.S. (2018). Exploring blockchain technology and its potential applications for education. Smart Learning Environments, 5(1): 1. https://doi.org/10.1186/s40561-017-0050-x

[8] Grech, A., Camilleri, A.F. (2017). Blockchain in Education. Publications Office of the European Union.

[9] Urien, P. (2021). Innovative countermeasures to defeat cyber attacks against blockchain wallets. In 2021 5th Cyber Security in Networking Conference (CSNet), Abu Dhabi, United Arab Emirates. https://doi.org/10.1109/CSNet52717.2021.9614649

[10] Conti, M., Kumar, E.S., Lal, C., Ruj, S. (2018). A survey on security and privacy issues of bitcoin. IEEE Communications Surveys & Tutorials, 20(4): 3416-3452. https://doi.org/10.1109/COMST.2018.2842460

[11] ConsenSys Diligence. (2021). Secure wallet design and best practices. https://consensys.net/diligence.

[12] Bano, S., Sonnino, A., Al-Bassam, M., Azouvi, S., McCorry, P., Meiklejohn, S., Danezis, G. (2020). SoK: Consensus in the age of blockchains. In AFT '19: Proceedings of the 1st ACM Conference on Advances in Financial Technologies, pp. 183-198. https://doi.org/10.1145/3318041.3355458

[13] Eyal, I. (2015). The miner's dilemma. In 2015 IEEE Symposium on Security and Privacy, San Jose, CA, USA, pp. 89-103. https://doi.org/10.1109/SP.2015.13

[14] Gencer, A.E., Basu, S., Eyal, I., Van Renesse, R., Sirer, E.G. (2018). Decentralization in bitcoin and Ethereum networks. In International Conference on Financial Cryptography and Data Security, Springer Berlin Heidelberg, pp. 439-457. https://doi.org/10.1007/978-3-662-58387-6_24

[15] Yli-Huumo, J., Ko, D., Choi, S., Park, S., Smolander, K. (2016). Where is current research on blockchain technology?—A systematic review. PloS One, 11(10): e0163477. https://doi.org/10.1371/journal.pone.0163477

[16] Sharples, M., Domingue, J. (2016). The blockchain and kudos: A distributed system for educational record, reputation and reward. In European Conference on Technology Enhanced Learning, Cham: Springer, pp. 490-496. https://doi.org/10.1007/978-3-319-45153-4_48

[17] Vasek, M., Thornton, M., Moore, T. (2014). Empirical analysis of denial-of-service attacks in the bitcoin ecosystem. In Financial Cryptography and Data Security. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-44774-1_5

[18] Huynh, T.T., Nguyen, T.D., Tan, H. (2019). A survey on security and privacy issues of blockchain technology. In 2019 International Conference on System Science and Engineering (ICSSE), 2019 International Conference on System Science and Engineering (ICSSE), pp. 362-367. https://doi.org/10.1109/ICSSE.2019.8823094

[19] Eyal, I., Sirer, E.G. (2014). Majority is not enough: Bitcoin mining is vulnerable. In International Conference on Financial Cryptography and Data Security, Springer Berlin Heidelberg, pp. 436-454. https://doi.org/10.1007/978-3-662-45472-5_28

[20] Daian, P., Goldfeder, S., Kell, T., Li, Y., Zhao, X., Bentov, I., Breidenbach, L., Juels, A. (2019). Flash boys 2.0: Front running, transaction reordering, and consensus instability in decentralized exchanges. arXiv preprint arXiv:1904.05234. https://doi.org/10.48550/arXiv.1904.05234

[21] Eskandari, S., Moosavi, S., Clark, J. (2019). Sok: Transparent dishonesty: Front-running attacks on blockchain. In International Conference on Financial Cryptography and Data Security, Cham: Springer, pp. 170-189. https://doi.org/10.1007/978-3-030-43725-1_13

[22] Flashbots. (2021). Flashbots: Protecting Ethereum from MEV. https://docs.flashbots.net/.

[23] Castro, M., Liskov, B. (1999). Practical byzantine fault tolerance. OsDI, 99: 173-186.

[24] Sousa, J., Bessani, A., Vukolic, M. (2018). A byzantine fault-tolerant ordering service for the hyperledger fabric blockchain platform. In 2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Luxembourg, pp. 51-58. https://doi.org/10.1109/DSN.2018.00018

[25] Kiayias, A., Russell, A., David, B., Oliynykov, R. (2017). Ouroboros: A provably secure proof-of-stake blockchain protocol. In Annual International Cryptology Conference, Cham: Springer, pp. 357-388. https://doi.org/10.1007/978-3-319-63688-7_12

[26] Saleh, F. (2021). Blockchain without waste: Proof-of-stake. The Review of Financial Studies, 34(3): 1156-1190. https://doi.org/10.1093/rfs/hhaa075

[27] Bentov, I., Gabizon, A., Mizrahi, A. (2016). Cryptocurrencies without proof of work. In International Conference on Financial Cryptography and Data Security, Springer Berlin Heidelberg, pp. 142-157. https://doi.org/10.1007/978-3-662-53357-4_10

[28] Bonneau, J. (2016). Why buy when you can rent? Bribery attacks on bitcoin-style consensus. In International Conference on Financial Cryptography and Data Security, Springer Berlin Heidelberg, pp. 19-26. https://doi.org/10.1007/978-3-662-53357-4_2

[29] Apostolaki, M., Zohar, A., Vanbever, L. (2017). Hijacking bitcoin: Routing attacks on cryptocurrencies. In 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, pp. 375-392.

https://doi.org/10.1109/SP.2017.29
[30] Heilman, E., Kendler, A., Zohar, A., Goldberg, S. (2015). Eclipse attacks on Bitcoin's peer-to-peer network. In 24th USENIX Security Symposium (USENIX Security 15), pp. 129-144.
[31] Marcus, Y., Heilman, E., Goldberg, S. (2018) Low-resource eclipse attacks on Ethereum's peer-to-peer network. IACR Cryptology.
[32] Nayak, K., Kumar, S., Miller, A., Shi, E. (2016). Stubborn mining: Generalizing selfish mining and combining with an eclipse attack. In 2016 IEEE European Symposium on Security and Privacy (EuroS&P), Saarbruecken, Germany, pp. 305-320. https://doi.org/10.1109/EuroSP.2016.32
[33] Rosenfeld, M. (2011). Analysis of bitcoin pooled mining reward systems. arXiv preprint arXiv:1112.4980. https://doi.org/10.48550/arXiv.1112.4980
[34] Decker, C., Wattenhofer, R. (2014). Bitcoin transaction malleability and MtGox. In European Symposium on Research in Computer Security, Cham: Springer International Publishing, pp. 313-326. https://doi.org/10.1007/978-3-319-11212-1_18
[35] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.
[36] Douceur, J.R. (2002). The sybil attack. In International Workshop on Peer-To-Peer Systems, Springer Berlin Heidelberg, pp. 251-260. https://doi.org/10.1007/3-540-45748-8_24
[37] Gervais, A., Karame, G.O., Wüst, K., Glykantzis, V., Ritzdorf, H., Capkun, S. (2016). On the security and performance of proof of work blockchains. In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna Austria, pp. 3-16. https://doi.org/10.1145/2976749.2978341
[38] Zhang, R., Xue, R., Liu, L. (2022). Security and privacy on blockchain. ACM Computing Surveys, 52(3), 1-34. https://doi.org/10.1145/3316481
[39] Lindell, Y. (2021). Fast secure two-party ECDSA signing. Journal of Cryptology, 34(4): 44. https://doi.org/10.1007/s00145-021-09409-9
[40] Alam, T. (2022). Blockchain-based educational certificates: A comprehensive study. Education and Information Technologies, 27, 12345-12365. https://doi.org/10.1007/s10639-021-10732-1
[41] Turkanović, M., Hölbl, M., Košič, K., Heričko, M., Kamišalić, A. (2018). EduCTX: A blockchain-based higher education credit platform. IEEE Access, 6: 5112-5127. https://doi.org/10.1109/ACCESS.2018.2789929
[42] Xie, J., Tang, H., Huang, T., Yu, F.R., Xie, R., Liu, J., Liu, Y. (2019). A survey of blockchain technology applied to smart cities: Research issues and challenges. IEEE Communications Surveys & Tutorials, 21(3): 2794-2830. https://doi.org/10.1109/COMST.2019.2899617
[43] Zheng, Z., Xie, S., Dai, H., Chen, X., Wang, H. (2017). An overview of blockchain technology: Architecture, consensus, and future trends. In 2017 IEEE International Congress on Big Data (BigData Congress), Honolulu, HI, USA, pp. 557-564. https://doi.org/10.1109/BigDataCongress.2017.85

**NOMENCLATURE**

| | |
|---|---|
| DLT | Distributed Ledger Technology |
| P2P | Peer-to-Peer |
| dApps | Decentralized Applications |
| DeFi | Decentralized Finance |
| NFT | Non-Fungible Token |
| PoW | Proof-of-Work |
| PoS | Proof-of-Stake |
| PBFT | Practical Byzantine Fault Tolerance |
| BFT | Byzantine Fault Tolerance |
| API | Application Programming Interface |
| BGP | Border Gateway Protocol |
| DNS | Domain Name System |
| DoS | Denial-of-Service |
| DDoS | Distributed Denial-of-Service |
| MITM | Man-in-the-Middle |
| MEV | Miner Extractable Value |
| NFC | Near Field Communication |
| EVM | Ethereum Virtual Machine |
| SMT | Satisfiability Modulo Theories |
| GDPR | General Data Protection Regulation |
| 2FA | Two-Factor Authentication |
| MFA | Multi-Factor Authentication |
| UID | Unique Identifier |
| HSM | Hardware Security Module |
| ASIC | Application-Specific Integrated Circuit |