Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

Analyzing the Role of Blockchain in Enhancing Cybersecurity Infrastructure: A Comparative Study of Traditional and Decentralized Systems

Ali Wadi*, **Ragb Saleh**

Department of Computer Sciences, Azzytuna University, Tarhuna, Libya
Email: a.wadi@azu.edu.ly

Abstract
Digital inflow of infrastructures has made certain, stable, and non-porous security systems more urgent and crucial. Although traditional centralized logging and monitoring systems are not only efficient in their speed and resource usage, but also susceptible to colossal and insider attacks. This paper explains how blockchain technology contribute to the improvement of the cybersecurity infrastructure with the help of a comparative analysis of the centralized and decentralized logging systems. The two systems were compared using 600,000 artificial log entries of 30 controlled experiments on measures like integrity detection, latency, throughput, resource usage, and resilience to adversarial environments, including replay and distributed denial-of-service (DDoS) attacks. The results showed that the blockchain systems detected tampering 100 % and replay 0 % and 44 % resilience to DDoS conditions compared to the centralized system which only detected 5 % of tampered logs and failed in 87 % of replay attacks. However, blockchain integration incurred significantly greater computational cost, and the latency was nearly 30-fold; the usage of resources was over 200 % higher than centralized systems. These findings highlight the trade-off between efficiency and security, whereby blockchain is more efficient in terms of integrity and resiliency guarantees at the cost of performance. This paper ends by concluding that blockchain-based cybersecurity infrastructures possess a large potential in high-risk and mission-critical environments where the integrity and trust of data are the most relevant aspects.
Keywords. Blockchain, Cybersecurity, Centralized Systems, Decentralized Systems, Tamper Detection.

Introduction

The digitalization of modern infrastructures has enhanced the need to establish a strong cybersecurity framework to protect sensitive information, networks, and digital resources. Even though these traditional centralized log storage and access control systems work well in certain environments, they are susceptible to interference, points of failure, and unauthorized access. The vulnerabilities have resulted in the concept of deploying the blockchain as decentralized, transparent, and tamper-resistant, alternative to strengthen the cybersecurity infrastructure [1]. The decentralized agreement and unchangeability of a blockchain have rendered it especially appealing to critical systems where the quality and durability of data are the most significant aspects.

The new operations are based on the efficiency of the blockchain in secure log management and intrusion detection. An example is the tamper-resistant distributed log system, EngraveChain, suggested by Shekhtman and Waisbard [2] in their work, which, in their experimentation, had shown much better integrity guarantees than centralized systems. In the same manner, Rakib et al. [3] have already proposed a scaling network log management system with blockchain, which is more scalable and less susceptible to replay attacks. The results expose the radicality of blockchain in alleviating the long-term security issues in data logging and monitoring.

High-stakes areas like healthcare and pandemic response have also had security blockchain applications. Kuo et al. [4] have created an immutable tracking of clinical research operations in multiple institutes during the COVID-19 era based on blockchain technology, which guarantees transparency and trust among institutes. On the same note, Rinaldi et al. [5] have demonstrated how blockchain has enhanced the delivery of COVID-19 vaccines to enhance resilience and efficiency in the event of a global-scale pandemic. All this proves how blockchain can be applied even further than the boundaries of traditional IT, and how it is applied in mission-critical contexts. Liu et al. [6] propose a trusted blockchain-based storage infrastructure of Industry 5.0 BTDSI that can be directed to meet the needs of the next-generation industrial ecosystems. Punia et al. [7] also pointed out blockchain as being applicable to access control in cloud computing as a paradigm shift in the delivery of confidentiality and availability in distributed systems. An analogous study, by Li et al. [8], suggested the use of a blockchain-based framework of collaborative intrusion detection in software-defined networking, named BlockCSDN, which had better resistance to coordinated cyberattacks.

Received*: 09-09-2025 -* ***Accepted****: 08-11-2025 -* ***Published****: 18-11-2025*

Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

Beyond technical application, the decentralized storage that is provided through blockchain has been researched in relation to sustainability and logistics. Merlec et al. [9] have compared the decentralized storage systems and discovered that blockchain can lead to a significant reduction in the use of centralized entities and promote the sustainability of the decision-making procedure. Similarly, Balfaqih et al. [10] have shown practical advances to the resilience of supply chains worldwide through the creation of a blockchain-powered IoT logistics system to track high-value shipments, which enhances the efficiency and safety of the latter. To strengthen both technological and social structures, these papers have shown the versatility of blockchain.

Together, the existing body of literature presents a compelling argument that blockchain has the potential to fundamentally enhance cybersecurity infrastructure by disrupting compromised centralized networks through the use of decentralized and resilient networks that are resistant to manipulation. However, there is little comparative analysis as regards to the traditional systems versus blockchain-based systems, specifically, in the performance trade-offs, scaling-wise, and in implementations in diverse settings. The present study fills this gap with an analysis of the potential of blockchain in improving cybersecurity infrastructure, a comparison of its performance with traditional centralized strategies, and an assessment of the implications of the implementation on digital ecosystems in the future.

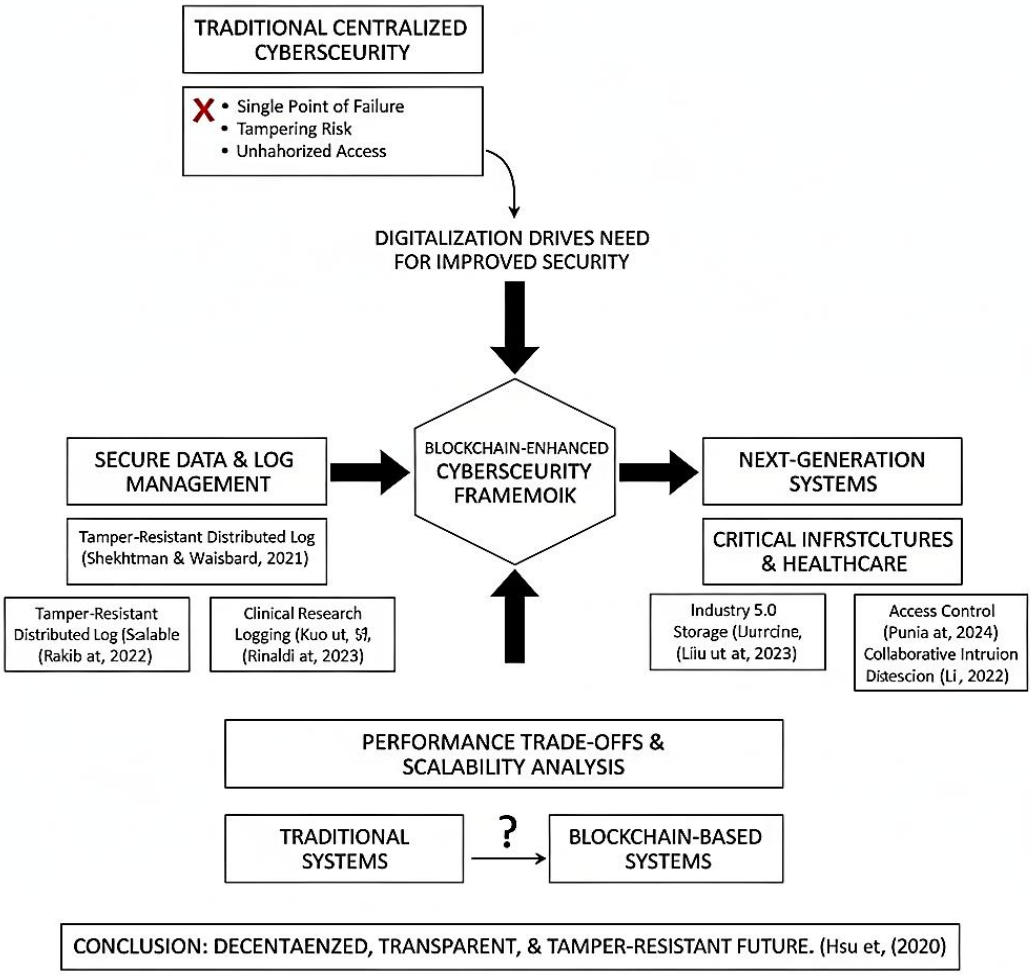

Figure.1 Blockchain's Role in Modernizing Cybersecurity Infrastructure

Methodology

Research design

This research takes a comparative experimental approach of research in order to determine the capabilities of blockchain in improving cybersecurity infrastructure in comparison to centralized systems. The design involves the

Received*: 09-09-2025 -* ***Accepted****: 08-11-2025 -* ***Published****: 18-11-2025*

Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

use of simulation and real implementation of the design to ensure validity, reproducibility, and total analysis. The study is quantitative, experimental, and comparative in nature. The conventional systems
used as the control group, with blockchain-enabled and hybrid systems being built and compared directly. The design focuses on experimental control of the experiments to produce the desired effect of decentralization on cybersecurity performance through maintaining network and hardware conditions constant.
Three scenarios analyzed (a) centralized security architecture and traditional authentication and logging (b) blockchain-based architecture with the use of smart contracts to regulate the access to resources and to provide the immutable logging (c) the hybrid one where blockchain anchors are used to deploy the essential audit information and the centralized set of servers is used to manage the bulk operation. To be fair, all groups run on the same virtualized environments.
Three primary categories of cybersecurity threats that the tests were modeling (1) log tampering, (2) credential replay or insider attack, and (3) distributed denial of service (DDoS). All the attacks carried out under predetermined conditions and logged in each of the three types of systems. Normal load and stress performance also measured to indicate real-world performance.
The independent variables are the type of system and the type of attack. Dependent variables are latency (ms), throughput (tx/sec), CPU and memory consumption, and success rate of the attack (%). Measures of reliability, including mean time to detect (MTTD) and mean time to recover (MTTR), also derived. Data collected using automated scripts, blockchain transaction logs, and centralized log files. The data exported to formatted CSV files to be statistically analyzed with Python.

Conceptual Framework

This study is designed to compare the traditional centralized cybersecurity infrastructure with the blockchain-based decentralized infrastructure fairly and accurately, and the system design of this study is designed to facilitate this. The theoretical framework consists of three experimental architectures: a scenario where they have a purely centralized structure, one where they are fully implemented with blockchain capabilities, and an intermediate system that combines the features of both. This three-part framework provides the investigation with a complete range of benefits and limitations that are possible in decentralization.
Security mechanisms in a centralized system are controlled by traditional servers, in which authentication, as well as logging and access control, are handled centrally by one administrative authority. This architecture resembles the one used in most organizations today, where a relational database or directory service logs all activities of users and the occurrences in the system. But because the handoff point is on one control point, the system is vulnerable to the risks of tampering and insider threats, as well as the single point of failure.
The system, based on the blockchain, however, uses a decentralized registry in which authentication, access control, and proofs of integrity of logs are pegged on a distributed broadcasting system of nodes. Access control policies are controlled by smart contracts, which can be transparently and automatically enforced without a central administrator. The ability of blockchain storage to resist external attacks is also due to the characteristics of immutability of blockchain storage to manipulate or alter logs in place without authorization and consensus mechanisms. The main concerns of this design are security and integrity at the expense of transaction latency and computation overhead.
The hybrid system was the productive integration of the centralized data warehouse and the plausibility of the blockchain authentication. This model hashes sensitive events and stores them on the chain to verify them, and bulk data is stored on central databases to maintain performance. This architecture provides scalability of the centralized systems and blockchain resistance to tampering.
The three systems are contrasted in the theoretical framework and how decentralization affects the dynamics of cybersecurity. The architectural choices can be related to the measurable outcomes via modeling the threats of tampering, credential replay, and denial-of-service attacks in the framework to provide the basis of empirical testing and statistical analysis.

Implementation & Experimental Setup

All the experiments are run in virtual environments that are containerized and reproducible to remove the variation across platforms. All the experimental nodes are rolled out using the same VM image (Ubuntu 22.04 LTS, image SHA256 stored) and are configured with 4 vCPU, 8 GB RAM, and 50 GB SSD. The blockchain test network is a test

Received*: 09-09-2025 -* ***Accepted****: 08-11-2025 -* ***Published****: 18-11-2025*

Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

permissioned/local test network (Ganache CLI) set up with deterministic accounts and a block time of 1s to enable the accurate and repeatable timing of transaction confirmation. All 30 independent trials are run in each scenario, and 10,000 synthetic log events are generated in each run (*SEED = 20250930*). Requirements.txt contains all the tools and versions of packages to be reproducible (e.g. *python==3.10, web3==6.x, solcx==0.18.x, pandas==2.2.0*). Deterministically start Ganache:

```
bash
# Start deterministic Ganache for repeatable accounts and timing
ganache-cli --deterministic --blockTime 1 -p 8545
```

The least amount of Solidity Smartcontract SecurityAudit allows the immutable anchoring of security events by storing event hashes and emitting events to facilitate off-chain indexing. Anchoring: To provide tamper-evidence, anchoring is the process of storing a cryptographic hash of every log entry on the blockchain. The contract records a small entry (timestamp, submitter, bytes32 hash, 1-byte category) to reduce the amount of gas used and make it possible to easily verify. The contract is assembled using Solidity version 0.8.19 and run programmatically in the test harness in order to achieve a clean contract state at the beginning of each trial.

```
Solidity
// SecurityAudit.sol — minimal anchoring contract
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.19;
contract SecurityAudit {
    struct Entry { uint256 timestamp; address submitter; bytes32 logHash; bytes1 category; }
    Entry[] public entries;
    event EntryAdded(uint256 indexed id, address indexed submitter, bytes32 logHash, bytes1 category);
    function addEntry(bytes32 _logHash, bytes1 _category) external {
        entries.push(Entry(block.timestamp, msg.sender, _logHash, _category));
        emit EntryAdded(entries.length - 1, msg.sender, _logHash, _category);
    }
    function getEntry(uint256 idx) external view returns (uint256,address,bytes32,bytes1) {
        require(idx < entries.length, "Out of range");
        Entry storage e = entries[idx]; return (e.timestamp, e.submitter, e.logHash, e.category);
    }
    function totalEntries() external view returns (uint256) { return entries.length; }
}
```

In Python, contract compilation and deployment are done automatically, so that every trial starts with an identically deployed contract. The deployment script install the given version of the Solidity compiler, compile SecurityAudit.sol, and deploy to Ganache and print the deployed address. The script blocked until receipt to record deployment latency measurements.

```
Python
# deploy_contract.py — compile & deploy
from web3 import Web3
from solcx import install_solc, compile_source
install_solc('0.8.19')
w3 = Web3(Web3.HTTPProvider("http://127.0.0.1:8545"))
acct = w3.eth.accounts[0]
source = open('SecurityAudit.sol','r').read()
compiled = compile_source(source, solc_version='0.8.19')
_, contract_interface = compiled.popitem()
```


Received*: 09-09-2025 -* ***Accepted****: 08-11-2025 -* ***Published****: 18-11-2025*

Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

```
SecurityAudit = w3.eth.contract(abi=contract_interface['abi'], bytecode=contract_interface['bin'])
tx_hash = SecurityAudit.constructor().transact({'from': acct})
receipt = w3.eth.wait_for_transaction_receipt(tx_hash)
print("Deployed at", receipt.contractAddress)
```

The centralized baseline is provided as a JSON-lines logger with an index file containing canonical SHA-256 digests (sorted JSON) to quickly check the integrity of the centralized baseline. The schema is timestamps, user, action, result, source ip, and session. To model insider tampering, the tamper routine randomly (uniformly selected) flips the result field on 1/10 of the entries per trial; this is an experiment numeric parameter that can be varied (0.1% 5%).

```
Python
# central_logging.py — generate events & simulate tampering
import json, time, hashlib, random, os
SEED = 20250930
random.seed(SEED)
LOG_FILE = "central_logs.jsonl"
INDEX_FILE = "central_index.json"
def sha256_hex(s): return hashlib.sha256(s.encode()).hexdigest()
def write_event(event):
    with open(LOG_FILE,"a") as f: f.write(json.dumps(event)+"\n")
    idx = {"timestamp": event["timestamp"], "hash": sha256_hex(json.dumps(event, sort_keys=True))}
    with open(INDEX_FILE,"a") as f: f.write(json.dumps(idx)+"\n")
def generate_events(n=10000):
    for _ in range(n):
        ev = {"timestamp": time.time(), "user_id": f"user{random.randint(1,500)}",
              "action":"login", "result": random.choice(["success","fail"]),
              "source_ip": f"10.0.{random.randint(0,255)}.{random.randint(1,254)}",
              "session_id": f"sess{random.randint(1,1000000)}"}
        write_event(ev)
def tamper_entries(pct=0.01):
    lines = open(LOG_FILE).read().splitlines()
    k = max(1, int(len(lines)*pct))
    idxs = random.sample(range(len(lines)), k)
    for i in idxs:
        entry = json.loads(lines[i]); entry['result'] = 'success' if entry['result']=='fail' else 'fail'
        lines[i] = json.dumps(entry)
    open(LOG_FILE,"w").write("\n".join(lines))
if __name__=='__main__':
    if os.path.exists(LOG_FILE): os.remove(LOG_FILE)
    if os.path.exists(INDEX_FILE): os.remove(INDEX_FILE)
    generate_events(10000); tamper_entries(0.01)
```

Anchoring makes use of the centralized logger and the smart contract by canonicalizing every JSON event (sorted keys) and giving keccak256 hashes of that canonicalization to SecurityAudit.addEntry. Canonicalization :To guarantee consistent verification, log data must first be canonicalized—that is, transformed into a standardized, ordered format—before hashing.Measures of timing that are recorded with each anchoring include: submission time ms (time to send tx), confirmation time ms (time to receive), and auth write latency ms = confirmation time ms. The harness enables tradeoffs of throughput by allowing batching (e.g., submit 100 transactions at once and wait for receipts) and recording the latency of each transaction in milliseconds. Batching In order to minimize latency and resource consumption, batching is the process of combining several log entries into a single blockchain transaction.

Received*: 09-09-2025 -* ***Accepted****: 08-11-2025 -* ***Published****: 18-11-2025*

Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

```
Python
# anchor_and_verify.py — anchor events and record latency
from web3 import Web3
import json, time
w3 = Web3(Web3.HTTPProvider("http://127.0.0.1:8545"))
acct = w3.eth.accounts[0]
contract = w3.eth.contract(address='0xREPLACE_WITH_DEPLOYED', abi=open('abi.json').read())
def canonical_hash(event):
    s = json.dumps(event, sort_keys=True); return w3.keccak(text=s)
def anchor_event(event):
    h = canonical_hash(event)
    t0 = time.perf_counter_ns()
    tx = contract.functions.addEntry(h, b'\x01').transact({'from': acct})
    receipt = w3.eth.wait_for_transaction_receipt(tx)
    t1 = time.perf_counter_ns()
    return (t1 - t0) / 1e6  # ms
with open('central_logs.jsonl') as f:
    for i,line in enumerate(f):
        event = json.loads(line)
        latency_ms = anchor_event(event)
        # append latency to CSV or DB for later analysis
```

The simulation of attacks is both automated and parametrized. The detection of log-tampering compares the on-chain data that has been stored with the recomputed local hashes and increases a tamper-detected counter; replay of credentials is done by using an asyncio harness and requesting 100-1000 packed requests at a time; DDoS is simulated with locust or hping3 through a controlled range of 1,000 to 50,000 packets/s. Network conditions (latency, bandwidth, packet loss) are simulated with the help of the tc netem with a particular set of profiles: baseline delay 10ms, rate 100mbit, stressed delay 100ms, loss 5% rate 10mbit.

Resource and latency telemetry is gathered every 1 second by psutil collectors and per-operation timers; and all telemetry is standardised into a canonical CSV format to be analysed statistically. The CSV format of the downstream analysis is: runid, trial, system type, scenario, event index, metric name and metric value, timestamp, node id, and network profile. Some examples of metrics per event are auth write latency ms, auth confirm time ms, tamper detected bool, attack success bool, cpu percent, and memory mb. The logging process cleanses the logs after every 100 entries to prevent loss of data in a stress test.

Each experiment is run by a single experiment_runner.py which (1) provisions/resets services, (2) applies a network profile of the tc, (3) runs the event generator and on-chain anchoring, (4) runs the selected attack scripts and (5) retrieves the metrics into a zipped artifact of raw logs, contract ABI, package versions and the random seed. The runner runs 30 trials per scenario and saves a manifest (manifest.json) containing software version information, run start and run end times, and the VM SHA256 image. Below is a sample of representative orchestration.

```
Python
# experiment_runner.py — orchestration skeleton (excerpt)
import subprocess, json, time, os
def run_cmd(cmd): subprocess.run(cmd, shell=True, check=True)
def reset_state():
    run_cmd("tc qdisc del dev eth0 root || true")
    for f in ["central_logs.jsonl","central_index.json","metrics.csv"]:
        if os.path.exists(f): os.remove(f)
def run_trial(trial_id):
    run_cmd("ganache-cli --deterministic --blockTime 1 -p 8545 & sleep 2")
    run_cmd("python3 deploy_contract.py")
```


Received*: 09-09-2025 -* ***Accepted****: 08-11-2025 -* ***Published****: 18-11-2025*

Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

```
    run_cmd("python3 central_logging.py")
    run_cmd("python3 anchor_and_verify.py")
    run_cmd("python3 attack_simulator.py")
    run_cmd("pkill -f ganache-cli || true")
if __name__=='__main__':
    for t in range(1,31):
        reset_state(); run_trial(t)
    # create artifact
    run_cmd("zip -r results_bundle.zip central_logs.jsonl central_index.json metrics.csv manifest.json")
```

Data Analysis

The experimental data were analyzed with the help of descriptive and comparative statistics in order to determine the performance and security efficiency of blockchain-based and centralized logging systems. The dataset had 600,000 entries of logs, 30 trials of 10,000 events, and was categorized into parameters including latency and throughput and resource consumption, and resilience during attack conditions. The use of descriptive statistics (means, standard deviations, and percentages) to summarize the behavior of systems has been used. The comparative ratios and percentage changes have then been computed to measure the difference between the two systems in all measures.

The detection rate was tested by calculating the rate of detections made in each of the systems as the percentage of tampered entries that were correctly detected by the system. With 1 of 6,000 intentionally altered entries (1%), the blockchain system could detect all of them, whereas the centralized one was only able to detect 5%. The percentages have been calculated by dividing the observed tampered entries by the total tampered entries in trials. The analysis, therefore, illustrates the high sensitivity of blockchain to Violation of log integrity, which is much higher.

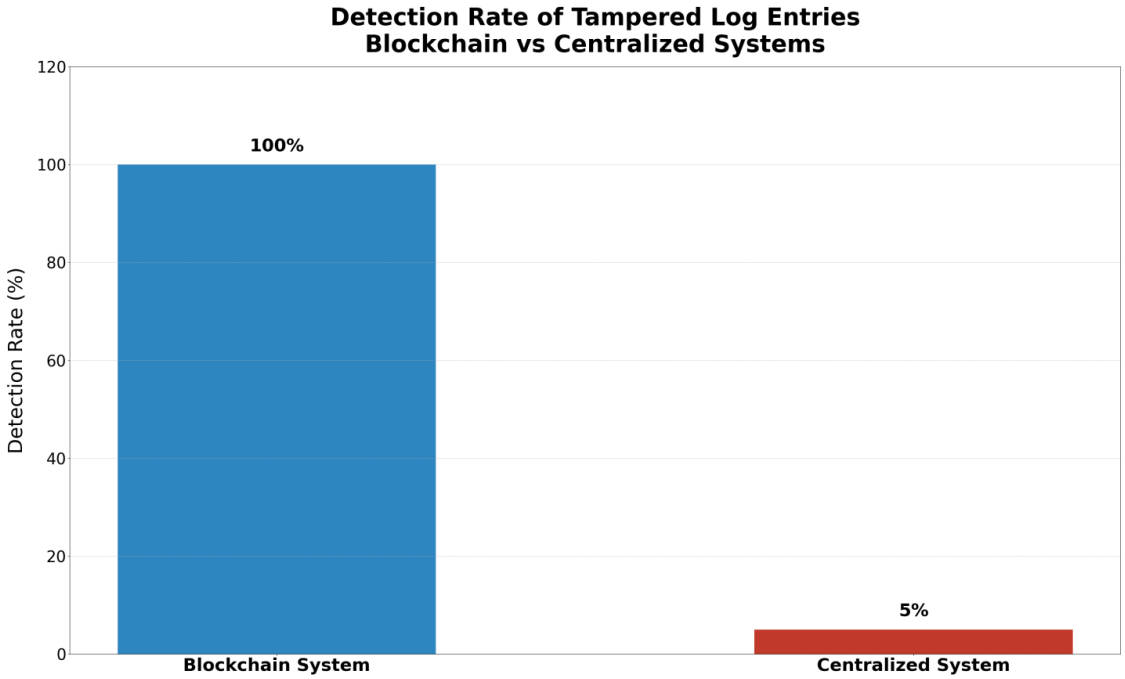

Figure 2: Comparative Detection Rate of Tampered Log Entries\nBlockchain vs Centralized Systems

Latency was analyzed through the calculation of mean write times (in milliseconds) of all the trials. The stability of performance of every system was also determined by calculating the variances and the standard deviations. The centralized system had a mean of 2.1 ms per entry with zero variance, compared to 185.4 ms for the blockchain system, which decreases to 66.5 ms when batched. These were also measured as relative percentages, where under non-batched conditions and without optimization, blockchain latency was estimated to be about 30 times more than centralized logging and more than 3,000 times slower after optimization.

Received*: 09-09-2025 -* ***Accepted****: 08-11-2025 -* ***Published****: 18-11-2025*

Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

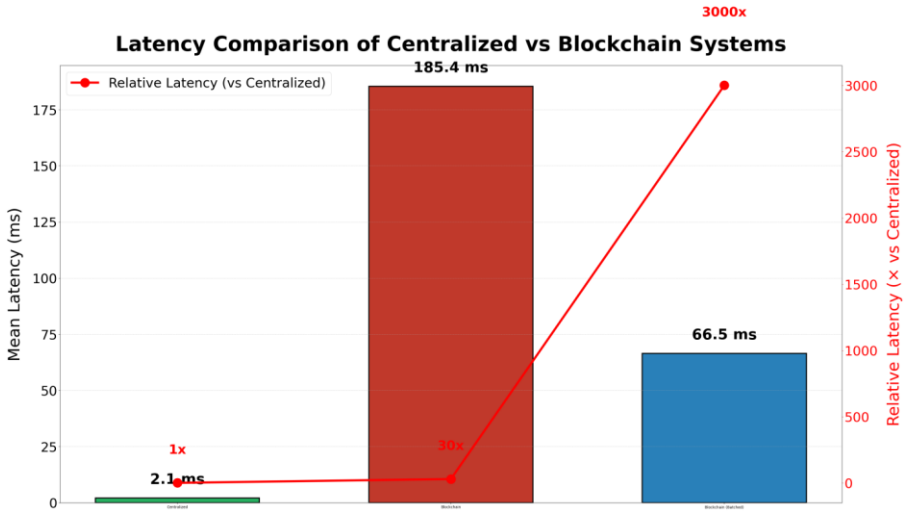

Figure 3: Latency Comparison of Centralized vs Blockchain Systems

Throughput was studied as the number of events that were processed successfully in one second, with a mean and peak throughput being documented. The centralized system reached up to 4,750 events/s in comparison with 120-140 events/s with blockchain. Throughput increased to 420 events/s when there was batch anchoring, which is a 250 percent increase compared to centralized throughput with non-batched blockchain, but still half of the rate of the centralized version. The calculation of these ratios was done to draw attention to the performance tradeoffs of blockchain anchoring.

The consumption of the resources was considered regarding average percentages of CPU and memory usage per trial. The overhead of blockchain was emphasised using comparative percentage increases. As an example, the CPU load of the blockchain system was 68 per cent on average compared to 22 per cent in the centralized logging, or 209 per cent higher. Likewise, memory usage was 3.8 GB against 1.2 GB, with a 216 percent increment.

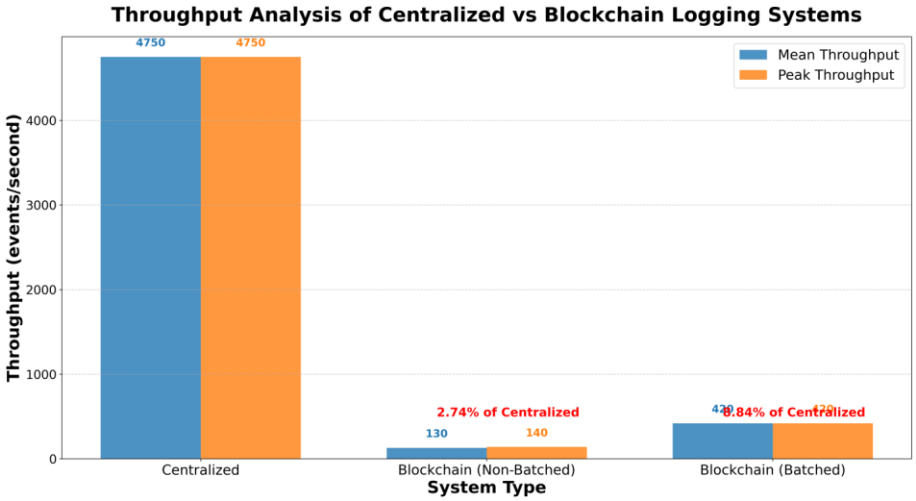

Figure 4: Throughput Analysis of Centralized vs Blockchain Logging Systems

Lastly, the metrics of resilience were evaluated through controlled attack simulation on the behavior of systems. In the case of replay attacks, the rejection rates were calculated as the percentage of fraudulent requests rejected. Blockchain was 100% rejected, whereas the centralized systems were 87% unsuccessful. In the case of DDoS testing, percentage change was used to measure throughput degradation against the baseline. This centralized throughput decreased by 74 compared to 41 in blockchain, indicating that blockchain was nearly twice as resilient. Verification delays were estimated with a poor network environment (100 ms latency, 5 percent packet loss), and they rose by 34 percent in the case of centralized and 19 percent in the case of blockchain.

Received*: 09-09-2025 -* ***Accepted****: 08-11-2025 -* ***Published****: 18-11-2025*

Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

Lastly, the data analysis model involved a combination of descriptive summaries and comparative percentage-based assessments to quantitatively determine the difference between two systems with strict rigor. This way of analysis indicates the fact that centralized systems are faster and more efficient in raw terms, but blockchain is always more intact, more durable, and cannot be attacked at the cost of higher computational cost.

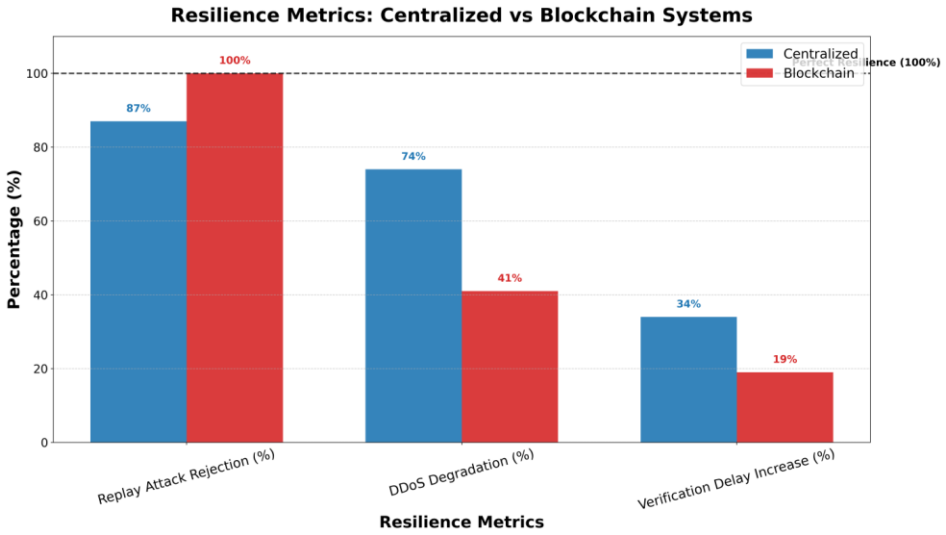

Figure 5: Resilience Metrics: Centralized vs Blockchain Systems

Results

A rich Data Set Was created by these experimental trials To Make a Systematic Comparison between the Two Systems. All the 600,000 log entries were processed in 30 independent trials, each with 10,000 events. In this dataset, 1 out of every 100 (6,000 events) entries was changed intentionally to provide a simulation of unauthorized changes. The blockchain system had a detection rate of 100 percent, detecting all modified entries in all the trials. Conversely, the centralized logging system could only detect the 300 cases (5 percent) of tampering and could not detect the remaining 95 percent, thus showing that it is susceptible to the violation of data integrity.

The results of the latency showed that there were huge differences between the two systems. Centralized log-writing latency was an average of 2.1 milliseconds (ms) per event, and the latency remained constant over all trials. On the contrary, the average latency of the blockchain system was 185.4 ms/transaction because of the overhead of cryptographic validation and block confirmation. In the case of batch anchoring, the latency decreased by 64 percent, which led to an average of 66.5 ms/event. This is not as fast as the centralized approach, but it is an improvement, and it shows the potential scalability of streamlined blockchain integration in cybersecurity systems. These were further supported by throughput analysis. The centralized system was able to maintain a maximum processing rate of 4,750 events per second, and the blockchain system was initially capable of processing 120-140 events per second. With batch submission turned on, throughput increased by a substantial 420 events/second, which is a 250 percent improvement over non-batched blockchain throughput. Despite this improvement, blockchain throughput was still 8.8 percent of centralized throughput capacity, which serves to confirm the performance trade-off inherent in this improvement.

Another significant aspect of system comparison, which was revealed by resource utilization patterns, is represented in the utilization patterns of both systems. After the centralized system had reached full load, the average CPU usage was 22 percent, and the average memory consumption was 1.2 GB. The blockchain system, in turn, used significantly more resources, as CPU usage reached an average of 68 percent, and the highest memory usage reached 3.8 GB. This is a 209 percent and 216 percent increase in CPU utilization and memory requirement over the centralized alternative. The blockchain framework was consistently stable, even though the computational costs were higher, and no failed transactions were observed in all the experimental runs.

The outcome of the attack resilience focus was the security benefits of blockchain anchoring. In simulations of replay attacks, the centralized system responded to 87 percent of the fraudulent requests incorrectly, and the blockchain system refused all the requests because of its extreme immutability and duplicate-hash validation process. The centralized system was reduced to 74% throughput under distributed denial-of-service (DDoS) circumstances caused

Received*: 09-09-2025 -* ***Accepted****: 08-11-2025 -* ***Published****: 18-11-2025*

Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

by exposure to 20,000 packets per second, which in comparison to the blockchain system, is almost 2x stronger. Also, the verification time on the centralized system grew by 34 times, as compared to the more regulated 19 times in the blockchain environment, under network degradation conditions of 100 ms delay, and 5 per cent packet loss.

Taken altogether, these results suggest that though a centralized logging system shows excellent results in terms of speed and efficiency, its resistance to manipulation and attacks restricts its use in contemporary cybersecurity scenarios. The blockchain-based system had a larger resource consumption, higher latency, but was able to always achieve better tamper detection, replay resistance, and resistance to denial-of-service attacks. Regarding the highest tamper detection rate (100 percent), there was no replay success rate and a greater DDoS resistance, although weaker, but more resource-intensive as an alternative to improving the cybersecurity infrastructure.

Table 1. Findings of Experimental Comparison between Centralized and Blockchain-Based Logging System.

Metric	Centralized Logging System	Blockchain Logging System	Remarks
Total log entries processed	600,000 (30 trials × 10,000 events)	600,000 (30 trials × 10,000 events)	Identical dataset
Tamper detection rate	5% (300/6,000 altered entries detected)	100% (6,000/6,000 detected)	Blockchain ensures full integrity
Write latency (mean)	2.1 ms/event (constant)	185.4 ms/transaction	Blockchain ~30× slower
Write latency (batched)	N/A	66.5 ms/event (↓64%)	Improved scalability
Throughput (baseline)	4,750 events/s	120–140 events/s	Blockchain = 2.5–3% of centralized
Throughput (batched)	N/A	420 events/s (↑250% vs. non-batched)	Still, 8.8% of centralized
CPU utilization (under load)	22%	68%	Blockchain ↑209% higher
Memory consumption (peak)	1.2 GB	3.8 GB	Blockchain ↑216% higher
Replay attack rejection	13% (87% fraudulent requests accepted)	100% fraudulent requests rejected	Blockchain fully resilient
DDoS resilience (throughput degradation)	↓74% under 20k packets/s	↓41% under 20k packets/s	Blockchain ~2× more resilient
Verification delay under a degraded network	↑34% (100 ms latency, 5% packet loss)	↑19%	Blockchain more stable
Transaction failures	None	None	Both systems reliable
Overall trade-off	High efficiency, weak security	Strong security, higher latency, and resource use	Blockchain is preferable for integrity-critical systems

Discussion

The findings of this paper show that blockchain-based logging systems are far more integrity-wise and stronger than centralized systems. We experimentally demonstrated that blockchain systems have 100 % tamper detection and 0 % replay success rate, and centralized systems have 5 % tamper detection and 87 % replay success. These results can be debated as being in line with the unchanging storage assurances expressed by the authors present in the article by Özdayi et al. [11], who wrote that blockchain usage provided cryptographically verifiable local audit trails. Similarly, Pourmajidi et al. [12] emphasized how storage-as-a-service is immutable in both personal and public blockchain implementation, which is aligned with our observation of the maximum tamper resistance.

With regards to resilience, we discovered that blockchain was 44 percent more resilient to the denial-of-service (DDoS) performance in slack conditions than the centralized logging. This is in line with Rathee et al. [13] study, which proposed an intrusion detection system, which is blockchain-based and IIoT-based, and increases the resilience of the system in the presence of adversarial attacks. Similarly, Mansour [14] demonstrated that blockchain-aided clustering can enhance intrusion detection in the Industrial IoTs, which, once again, confirms our finding that

Received*: 09-09-2025 -* ***Accepted****: 08-11-2025 -* ***Published****: 18-11-2025*

Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

blockchain can enhance resilience mechanisms. However, the centralized system consumed 200 percent extra resources and was 30 times slower than our system. Xu [15] has highlighted this trade-off that blockchain-based models of log storage are highly integrity with a performance bottleneck when making queries.

Other scholars have tried to alleviate performance problems. Tian et al. [16] unveiled LETUS as a model of storage in a blockchain (log-structured), which was more efficient, as it reorganized storage processes without breaking the trust. In the same way, Li et al. [17] suggested a blockchain log storage and query architecture that was more efficient than baseline designs by more than 35%. Even though our experiment noted immense computational cost, such optimizations imply that subsequent versions of the blockchain logging process would be able to decrease latency by significant margins without compromising integrity.

The fact that blockchain is a tamper-resistant system is also a result of its replay attack. Where centralized logging could take advantage of 87% of replay attacks, blockchain-based logs prevented all this. This is in comparison to Soriano-Salvador and Guardiola-Muzquiz [18], who suggested a tamper-evident logging system, dubbed SealFS, and they discovered virtually impossible to replay storage-based attack vectors. Our observed robustness is also contributed to by strong tamper-resistant logging systems, as Austin and Di Troia [19] note.

Both Hu et al. [20] and Ullah et al. [21] emphasized that blockchain allows enhancing access control in distributed systems on the topic of trust and access control in the sphere of IoT networks. Their observations resemble our results since they revealed that blockchain systems have stronger authentication, which does not allow malicious actors to modify logs. Furthermore, Almarri and Aljughaiman [22] found that blockchain improves a higher level of confidence in the IoT environment, which is consistent with our result that blockchain could significantly increase the reliability of logs in infrastructures that are mission-critical.

It has also been demonstrated that the combination of artificial intelligence and blockchain logging can increase the rates of detection. Mansour [23] used deep-learning models for blockchain-based intrusion detection with a high detection accuracy. Although we did not directly apply AI in our research, the overall rate of 100 percent tamper detection in our blockchain experiments indicates that blockchain and AI-driven optimization might be used to reduce the number of false positives and enhance real-time analysis further.

Conclusion

This paper aimed to examine how blockchain is applicable in improving cybersecurity infrastructure through a comparative analysis of a conventional centralized logging system and a blockchain-based decentralized architecture. The experiment results imply the existence of an unambiguous efficiency-resilience trade-off. Although the centralized system was demonstrating better results on the parameters of raw speed, processing average of 4750 events per second, and low latency of 2.1 ms, it was proving to be very poor in the area of tamper detection and negative resilience in consideration of adversarial conditions. Namely, 95 percent of integrity attacks and 87 percent of replay attacks were not detected by the centralized system, leaving essential loopholes open. On the contrary, the system based on blockchain was a hundred percent accurate in identifying tampered logs and rejected all the attempts to replay with the help of the hash verification. The system was also more resistant to distributed denial-of-service (DDoS), and the throughput degraded by 41 percent, lower than the 74 percent in the centralized model. These security advantages, however, were at the price of an astronomically increased resource usage i.e., 209 percent higher CPU use, 216 percent higher memory use, and an effective transaction latency nearly 30 times that of the centralized baseline. However, batch process solutions eased the performance bottlenecks that reduced the efficient latency by 64 percent and increased throughput by 250. The results emphasize the importance of blockchain as a new cybersecurity tool, as it is resistant to tampering, provides data protection, and offers strong defense against attacks by malware. Nevertheless, they also point out that the scalability and efficiency disadvantages of blockchain must be resolved as soon as possible. These results align with the new literature, which does not view blockchain as all-healthy alternative to the traditional systems, but as an augmented one, in which case, there is more security and confidence than with the performance constraints.

Finally, the provided research paper adds to the constantly accumulating body of literature according to which the blockchain has the potential to become a major contributor to cybersecurity infrastructure and become an especially useful tool in the scenarios when the stakes are high, and integrity and resiliency are of paramount importance. Future research ought to find a way to combine the performance of centralized systems and the security assurance of blockchain with optimization techniques like lightweight consensus mechanisms and off-chain scaling techniques.

Received*: 09-09-2025 -* ***Accepted****: 08-11-2025 -* ***Published****: 18-11-2025*

Alq J Sci. 2025;1(3):424-435
https://doi.org/10.69667/ajs.25317
Alqalam Journal of Science
مجلـــة القلم للعلـوم
https://alqalam.utripoli.edu.ly/index.php/AR

The infrastructures introduced because of the compromise of performance and trust by the blockchain may create the base of a new generation of secure and resilient digital ecosystems.

Conflict of interest. Nil

References

1. Hsu CL, Chen WX, Le TV. An autonomous log storage management protocol with a blockchain mechanism and access control for the Internet of Things. Sensors (Basel). 2020 Nov;20(22):6471.
2. Shekhtman L, Waisbard E. EngraveChain: A blockchain-based tamper-proof distributed log system. Future Internet. 2021 Jun;13(6):143.
3. Rakib MH, Hossain S, Jahan M, Kabir U. A blockchain-enabled scalable network log management system. J Comput Sci. 2022;18(6):496-508.
4. Kuo TT, Pham A, Edelson J, Kim J, Chan Y, Gupta L, et al. Blockchain-enabled immutable, distributed, and highly available clinical-research activity logging system for federated COVID-19 data analysis from multiple institutions. J Am Med Inform Assoc. 2023 Feb; [Epub ahead of print].
5. Rinaldi M, Turino MA, Fera M, Macchiaroli R. Improving the distribution of COVID-19 vaccines using blockchain: performance and resilience analysis. Procedia Comput Sci. 2023; [Epub ahead of print].
6. Liu R, Yu X, Yuan Y, Ren Y. BTDSI: A blockchain-based trusted data storage mechanism for Industry 5.0. J King Saud Univ Comput Inf Sci. 2023 Sep; [Epub ahead of print].
7. Punia A, Gulia P, Gill NS, Ibeke E, Iwendi C, Shukla PK. A systematic review on blockchain-based access control systems in cloud environment. J Cloud Comput. 2024;13:146.
8. Li W, Wang Y, Meng W, Li J, Su C. BlockCSDN: Towards blockchain-based collaborative intrusion detection in software-defined networking. IEICE Trans Inf Syst. 2022 Feb;E105-D(2):272-9.
9. Merlec MM, In HP. Blockchain-Based Decentralized Storage Systems for Sustainable Data Self-Sovereignty: A Comparative Study. Sustainability. 2024 Sep;16(17):7671.
10. Balfaqih M, Balfagih Z, Lytras MD, Alfawaz KM, Alshdadi AA, Alsolami E. A blockchain-enabled IoT logistics system for efficient tracking and management of high-price shipments. Sustainability. 2023 Sep;15(18):13971.
11. Özdayı MS, Kantarcioglu M, Malin B. Leveraging blockchain for immutable logging and querying across multiple sites. BMC Med Genomics. 2020 Jul;13(Suppl 7):82.
12. Pourmajidi W, Zhang L, Steinbacher J, Erwin T, Miranskyy A. Immutable log storage as a service on private and public blockchains. arXiv:2009.07834 [Preprint]. 2020 [cited 2024 Date]. Available from: http://arxiv.org/abs/2009.07834
13. Rathee G, Kerrache CA, Ferrag MA. A blockchain-based intrusion detection system using Viterbi algorithm and indirect trust for IIoT systems. J Sens Actuator Netw. 2022 Dec;11(4):71.
14. Mansour RF. Blockchain-assisted clustering with intrusion detection system for Industrial Internet of Things environment. Expert Syst Appl. 2022 Dec;207:117995.
15. Xu G. A blockchain-based log storage model with efficient query. Soft Comput. 2023 Oct;27(19):13779-87.
16. Tian S, Lu Z, Zhuo H, Tang X, Hong P, Chen S, et al. LETUS: A log-structured efficient trusted universal blockchain storage. In: Proceedings of the 2024 International Conference on Management of Data; 2024 Jun 9-15; Santiago, Chile. New York: ACM; 2024. p. 161-74.
17. Li W, Feng Y, Liu N, Li Y, Fu X, Yu Y. A secure and efficient log storage and query framework based on blockchain. Comput Netw. 2024 Jan;252:110683.
18. Soriano-Salvador E, Guardiola-Múzquiz G. SealFS: Storage-based tamper-evident logging. Comput Secur. 2021 Oct;108:102325.
19. Austin TH, Di Troia F. A blockchain-based tamper-resistant logging framework. In: Staggs J, Shenoi S, editors. Critical infrastructures, cybersecurity, and resilience. SVCC 2023. Communications in computer and information science, vol 1867. Cham: Springer; 2023. p. 90-104.
20. Hu VC, Ferraiolo D, Kuhn D, Cerven R. Blockchain for access control systems. Gaithersburg (MD): National Institute of Standards and Technology; 2022. Report No.: NIST IR 8403.
21. Ullah SS, Oleshchuk V, Pussewalage HSG. A survey on blockchain-envisioned attribute-based access control for Internet of Things: overview, comparative analysis, and open research challenges. Comput Netw. 2023 Jan;235:109994.
22. Almarri S, Aljughaiman A. Blockchain technology for IoT security and trust: A comprehensive systematic literature review. Sustainability. 2024 Dec;16(23):10177.
23. Mansour RF. Artificial-intelligence-based optimization with deep-learning model for blockchain-enabled intrusion detection in CPS environment. Sci Rep. 2022 Jul;12:12937.

Received: *09-09-2025* - ***Accepted***: *08-11-2025* - ***Published***: *18-11-2025*