International Journal of Advance and Innovative Research
Volume 12, Issue 2 (XXI): April - June 2025
ISSN 2394 - 7780

A SECURE AND SCALABLE VOTING SYSTEM USING CLOUD COMPUTING AND BLOCKCHAIN TECHNOLOGY

[1]Ankit Choubey, [2] Ashish Singh, [3] Deepika Sharma, [4]Vikas Kumar Yadav and [5]Dr. Pooja Kapoor
[1, 2, 3, 4]Department of CSE, Mangalmay Institute of Engineering and Technology, Greater Noida, Uttar Pradesh, India
[5]Research Coordinator & Professor, Mangalmay Institute of Engineering & Technology, Greater Noida, Uttar Pradesh, India

ABSTRACT
In the evolving landscape of information technology, the integration of cloud computing and blockchain in electronic voting systems has garnered considerable attention due to its potential to enhance scalability, security, cost-effectiveness, and accessibility. This paper explores the design, implementation, and security implications of a secure, cloud-based voting system. Built using Node.js and MongoDB, and incorporating blockchain for immutability, the system addresses critical factors such as voter authentication, transparency, and performance. Through simulation, testing, and performance analysis, our research demonstrates the feasibility of deploying cloud-based voting systems in real-world scenarios while overcoming traditional limitations.

Index Terms— *Voting System, Cloud Computing, Blockchain, Web Security, MongoDB, Node.js, E-Governance, Smart Contracts, Distributed Ledger, Election Integrity*

INTRODUCTION
Modern democracies rely heavily on secure and transparent voting mechanisms. Traditional voting systems often face challenges including logistical complexity, fraud potential, and limited accessibility. As digital infrastructure advances, there is a growing push to modernize electoral systems through cloud computing and blockchain integration. These technologies offer scalable, resilient, and secure solutions that address the limitations of paper-based and early electronic voting systems. Leveraging cloud platforms enables real-time processing, improved data management, and cost savings, while blockchain ensures vote integrity through immutability.

LITERATURE REVIEW
Traditional systems, such as paper ballots and in-person voting, face issues like human error, disenfranchisement, and fraud. Estonia's internet voting system is an early example of e-voting adoption, though not without controversy regarding security and transparency. Recent studies [2], [3] suggest cloud infrastructure enables real-time analytics, enhanced system availability, and streamlined voter access. Blockchain platforms such as Ethereum and Hyperledger have been proposed to secure vote records against tampering [5].

The convergence of cloud and blockchain has been widely explored. Works such as [2], [3] emphasize the role of data-driven and secure architectures in achieving transparency and reliability in decentralized systems.

SYSTEM ARCHITECTURE
The system is composed of five core components:

- **Frontend Interface:** User interaction portal.
- **Backend Services:** Node.js with Express.js handling business logic.
- **Database Layer:** MongoDB for persistent data storage.
- **Blockchain Module:** Ethereum-based smart contract integration.
- **Cloud Hosting:** AWS EC2 for hosting with load balancing and monitoring.

This microservices-based architecture ensures modularity, scalability, and independent testing of each component.

DESIGN AND IMPLEMENTATION

- **Data Schemas**: Mongoose schemas in Node.js define candidate and voter models.
- **Verification Module**: Responsible for collecting voter data, authenticating identities, and enforcing email uniqueness.

ISSN 2394 - 7780

- **Blockchain Smart Contracts:** Handles vote submissions and restricts duplicate voting via mappings.
- **API Infrastructure**: RESTful APIs created and tested using Postman for user, voting, and result operations.
- **Cloud Deployment:** Dockerized services are deployed on AWS EC2; MongoDB Atlas ensures reliable and high-availability data storage.

BLOCKCHAIN INTEGRATION

Blockchain integration is centered on a Solidity smart contract. Below is an excerpt demonstrating the prevention of double voting:

SOLIDITY

```
mapping(address => bool) public hasVoted;
function castVote(uint candidateId) public {
    require(!hasVoted[msg.sender], "Already voted");
    votes[candidateId]++;
    hasVoted[msg.sender] = true;
}
```

This logic enforces immutability and vote authenticity.

CLOUD COMPUTING ADVANTAGES

The cloud infrastructure provides:

- Dynamic scalability with elastic resource allocation
- End-to-end encryption using TLS
- Real-time analytics and tabulation
- Reduced operational cost
- Flexible deployment using containers and virtual machines Load balancers and container orchestration improve fault tolerance and system availability.

SECURITY CONSIDERATIONS

Key features include:

- Encrypted communication (TLS/HTTPS)
- JWT-based secure session management
- Input sanitation to prevent injection attacks
- Penetration testing of API endpoints
- Blockchain immutability ensuring verifiable voting trails

EVALUATION AND RESULTS

A. Simulation

Tested with 10,000 simulated voters.

B. Performance

Average API response time: 220 ms. Peak throughput: 750 transactions/sec.

C. Security

The system passed vulnerability scans and stress tests, confirming its robustness against attacks.

CHALLENGES AND LIMITATIONS

- No mobile application support in the current version
- MVP includes only partial blockchain deployment
- Privacy vs traceability dilemma
- MongoDB exhibited scaling limitations under heavy load

ISSN 2394 - 7780

FUTURE WORK

Future enhancements include:

- Full-scale blockchain deployment with consensus protocols
- Integration with national digital identity platforms
- Use of AI for anomaly and fraud detection
- Biometric authentication features
- Multilingual mobile-first user interfaces

CONCLUSION

This study demonstrates the viability of combining cloud computing and blockchain technologies for building secure and scalable e-voting systems. The proposed architecture and prototype offer promising results in terms of performance, security, and usability. Further developments can enable real-world adoption and redefine electoral systems in digital democracies

REFERENCES

[1] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008.

[2] O. Babaoglu, et al., "Towards Data-Driven Autonomics in Data Centers," 2015.

[3] M. Doğan, et al., "IDMoB: IoT Data Marketplace on Blockchain," 2018.

[4] A. Masarweh, et al., "The Legal Aspects of Cybersecurity in E-Voting," 2023.

[5] Ethereum Foundation, "Ethereum White Paper," [Online]. Available: https://ethereum.org/en/whitepaper/

[6] MongoDB Inc., "MongoDB Documentation," [Online]. Available: https://www.mongodb.com/docs/

[7] Amazon Web Services, "AWS Documentation," [Online]. Available: https://aws.amazon.com/

[8] V. Musale, et al., "Secure Electoral Voting System Using Homomorphic Encryption," 2024.

[9] P. Antunes, et al., "Data Privacy in Microservices," Springer, 2024.

[10] M.D. Moloja, "Cloud-based Intrusion Detection for Mobile Voting," 2018.

[11] M.A. Ferrag, et al., "Security and Privacy for Green IoT-Based Agriculture," IEEE, 2020.

[12] G. Zyskind, et al., "Decentralizing Privacy with Blockchain," in *Proc. IEEE SPW*, 2015.

APPENDICES APPENDIX A: POSTMAN SAMPLE OUTPUT

Appendix B: Schema Diagrams (To be Included)

- Verification Schema
- Candidate and Voter Models