Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
Vol. 29 (2026), No. 10-B, pp. 4407–4416
DOI : 10.47974/JDMSC-3061

Threshold cryptography for distributed ledger systems

Budigi Prabhakar [†]
Department of Physics
Noida International University
Greater Noida 203201
Uttar Pradesh
India

Takveer Singh *
Centre for Research Impact and Outcome
Chitkara University Institute of Engineering and Technology
Chitkara University
Rajpura 140401
Punjab
India

Neelabh Sao [§]
Department of Computer Science & Engineering
Rungta College of Engineering & Technology
Rungta International Skills University
Bhilai 490024
Chhattisgarh
India

Kothakonda Sairam [‡]
Centre for Multidisciplinary Research
Anurag University
Hyderabad 500088
Telangana
India

[†] *E-mail:* budigi.prabhakar@niu.edu.in
* *E-mail:* takveer.singh.orp@chitkara.edu.in (Corresponding Author)
[§] *E-mail:* neelabh.sao@rungta.org
[‡] *E-mail:* kothakondasairam.ks@proton.me

© Taru Publications

Sulabha Narendra Patil [@]
Department of Engineering Science and Humanities
Vishwakarma Institute of Technology
Pune 411037
Maharashtra
India

Rajendra Vasantrao Patil [#]
Department of Computer Engineering
Shri Shivaji Vidya Prasarak Sanstha's
Bapusaheb Shivajirao Deore College of Engineering
Dhule 424005
Maharashtra
India

Abstract

Threshold cryptography has emerged as a promising approach to enhance security, decentralization and fault tolerance in distributed ledger systems. The paper discusses the implementation of threshold cryptography schemes to blockchain systems to mitigate single-point key failures, and improve the authentication of transactions. The paper presents an architectural scheme that entails threshold key generation, distributed key management and collaborative signature techniques that are suitable for the wallets and validator nodes. Mathematical formulations of thresholds signatures are defined to demonstrate the appropriateness and strength in adverse scenarios. It exhibits a better security resilience, attack resistance and reliability of the system, as compared to the conventional cryptography methods. Experimental results are analyzed risk of key compromise is reduced and more distribution of trust is achieved without a reduction in efficiency. The proposed solution is particularly applicable to decentralized finance and the implementation of enterprise blockchain which offers a safe and scalable alternative to the existing distributed systems.

Subject Classification: *68M14, 68M25.*

Keywords: *Threshold cryptography, Distributed ledger, Threshold signatures, Blockchain security, Key management, Decentralized systems.*

1. Introduction

By eliminating the necessity to have central entities and trustless interactions between actors, decentralized systems and, in particular, blockchain technologies have changed the structure of secure, decentralized systems [1, 2]. Some of the many applications based on these systems include cryptocurrencies, decentralized finance, supply chain management and digital identity systems [3, 4]. However, despite the highly designed consensus mechanisms and

[@] *E-mail:* **sulbha.patil@vit.edu**
[#] *E-mail:* **patilrajendra.v@gmail.com**

cryptographic foundations, the conventional ledger systems are likely to have a single secret key that enables the authorization of transactions, which is extremely dangerous in terms of key compromise, insider assaults, and failure points [5, 6, 7]. The big agenda has been thresholds cryptography to deal with these concerns as a powerful paradigm towards distributed key control and collaborative security [8, 9]. A threshold scheme is implemented in such a way that a secret key is broken down into various shares that are shared with a group of participants in such a way that only a predefined group, which is referred to as the threshold, can reassemble the key or produce valid cryptographic operations [10, 11]. This would promote resilience by making sure that no one particular body is in full control of sensitive cryptographic content.

2. Distributed Ledger System

A DLS runs coordinated accounts on decentralized computers, which makes it both transparent and unchangeable and resilient to failure due to consensus algorithms, cryptographic authentication, and peer-to-peer networking solutions [12, 13]. Figure 1 demonstrates the parts of the distributed ledger system that allow processing the data securely and with decentralization.

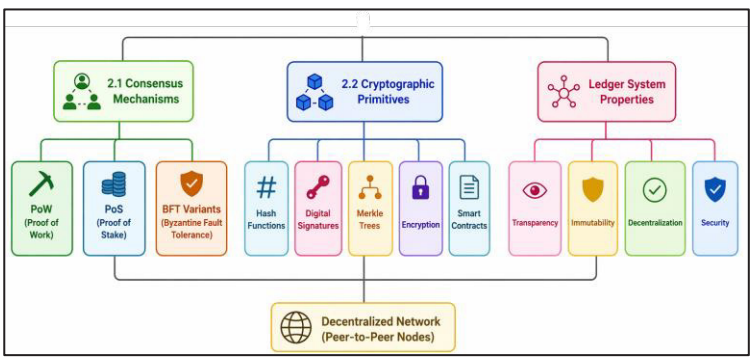


Figure 1

Distributed Ledger Systems Overview Using Simple Structured Components

2.1. *Consensus Mechanisms (PoW, PoS, BFT variants)*

Proof of Work, Proof of Stake and Byzantine fault tolerance are examples of consensus mechanisms that guarantee unanimity among the distributed nodes, and authenticate transactions safely without compromising on scalability, power economy and fault tolerance [13], [14].

$$H(B_i|n_i|) = sqrt\left(\left(\frac{(H(B_i|n_i|))^2}{D_i^2}\right)^2 + \int_0^1 \left(\frac{H(B_i|n_i|)*t + t^3}{D_i}\right)^2 dt\right) \quad (1)$$

The Proof-of-Work hash constraint in the equation (1) sets a computation requirement in which the computation output is required to meet a difficulty-bound formulation, which requires minerals to make intensive computations

prior to block approval. The presence of the terms of the form of the integral and squared terms, represent the complexity and resistance to the trivial solutions.

$$P_i = sqrt\left(\left(\frac{s_i}{\Sigma_{\{j=1\}}^{\{N\}s}}\right)^2 + \int_0^1\left(\left(\frac{s_i}{\Sigma_{\{j=1\}}^{\{N\}s}}\right) * t + t^3\right)^2 dt\right) \quad (2)$$

The Proof-of-Stake selection probability in equation (2) is the chances of a validator being selected according to its stake share.

$$R_i = sqrt\left((\alpha * s_i + \beta * T_i)^2 + \int_0^1\left(\left((\alpha * s_i + \beta * T_i) * t + t^2\right)^2\right) dt\right) \quad (3)$$

To ensure proportionality of incentives and balance in the system, the validator reward function in the equations (3) will have a combination of stake and participation time with weighted coefficients.

$$C_{final} = sqrt\left(\left(\Sigma_{\{i=1\}}^{\{N\}(P_i * v_i)}\right)^2 + \int_0^1\left(\left((P_i * v_i) * t + t^2\right)^2\right) dt\right) \quad (4)$$

Lastly, consensus validation expression in equation (4) is a weighted compilation of validator decisions, such that final consensus at the end of the consensus is a representation of collective participation, as well as consistency, security and integrity of distributed ledger nodes.

2.2. *Cryptographic Primitives in Blockchain Systems*

Cryptography primitives such as hash functions, digital signature, Merkle trees and encryption are also used in blockchain systems to provide integrity of data, authentication, data confidentiality and verification of transactions.

$$h(x) = sqrt\left((H(x)^2)^2 + \int_0^1((H(x) * t + t^3)^2)\, dt\right) \quad (5)$$

The hash construction inEq. (5) is a transformation that transforms the input data to a fixed-length digest that ensures the integrity and resistance to collisions. The higher-order and integral terms in it indicate complexity in computations and diffusion characteristics.

$$\sigma = sqrt\left(((H(m) + x * r \bmod q)^2) + \int_0^1\left(\left((H(m) + x * r \bmod q) * t + t^2\right)^2\right) dt\right) \quad (6)$$

To formulate digital signature, shown in the equation below (6), the message hashing is used along with the private key elements and the randomness, which generates a secure signature that ensures authenticity and non-repudiation. The cryptographic security is enhanced by the modular operation.

$$V = sqrt\left(\left(\left(g^{\{H(m)\}} * y^r \bmod p\right)^2\right) + \int_0^1\left(\left(\left(g^{\{H(m)\}} * y^r \bmod p\right) * t + t^3\right)^2\right) dt\right) \quad (7)$$

The authenticity concept in the signature verification expression in the form of Eq. (7) involves the comparison of the computed values over the public

parameters to verify authenticity so that only valid signatures will be accepted by the system.

$$E(m) = sqrt\left(((m^e mod\ N)^2) + \int_0^1 \left(\left((m^e mod\ N) * t + t^3 \right)^2 \right) dt \right) \quad (8)$$

$$D(c) = sqrt\left(((c^d mod\ N)^2) + \int_0^1 \left(\left((c^d mod\ N) * t + t^2 \right)^2 \right) dt \right) \quad (9)$$

Plaintext security in the encryption transformation of the form shown in Eq. (8) is ensured by a public exponent which transforms plaintext to ciphertext that cannot be accessed by an unauthorized party. Lastly, the original message is decrypted by the use of a private key in the decryption process of the message in the equation of (9) to ensure the correctness of the message and that it is confidential.

3. Integration of Threshold Cryptography in Ledger Systems

Threshold cryptography is implemented into ledger systems by sharing secret keys with individuals, providing collaborative security, avoiding single-point failures, and promoting trust, resilience and decentralized control features.

3.1. *Threshold Key Management for Wallets and Nodes*

Threshold key management separates private keys into several shares in wallets and nodes, and provides a secure storage, eliminating the possibility of unauthorized access, and recovery by reassembling the key collaboratively by participants.

$$f(x) = sqrt\left(\left(\left(a_0 + \Sigma^i_{\{i=1\}^{\{t-1\}}_i a_x} \right)^2 \right)^2 + \int_0^1 \left(\left(\left(a_0 + \Sigma^i_{\{i=1\}^{\{t-1\}}_i a_x} \right) * t + t^3 \right)^2 \right) dt \right) \quad (10)$$

The structured representation in the form of the secret key polynomial in Eq. (10) is a representation that allows the original secret to be encoded by the constant term and randomness by increasing the order of the coefficients, which allows distribution safety. The formulation aids in the reconstruction based on thresholds using a polymorphic evaluation.

$$s_j = sqrt\left((f(j)^2)^2 + \int_0^1 ((f(j) * t + t^2)^2)\, dt \right), j \in \{1,2,\dots,n\} \quad (11)$$

In Eq. (11), the shares are created by considering the evaluation of the different points of the polygraph in which the individual has a different piece of the key without any disclosure of the full key.

$$V_K = sqrt\left(((g^K mod\ p)^2)^2 + \int_0^1 (((g^K mod\ p) * t + t^3)^2)\, dt \right) \quad (12)$$

The important consistency check in the equation (12) is the one that checks the accuracy of the reconstructed key, by verifying it with public parameters, avoiding tampering with or invalid reconstruction.

$$K_{new} = sqrt\left(((K + \Delta K)^2)^2 + \int_0^1 \left(\left((K + \Delta K) * t + t^2 \right)^2 \right) dt \right) \quad (13)$$

Lastly, the secure key update scheme in Eq. (13) implements controlled variation of the key, which allows periodic update, without revealing underlying secret elements, and thus, increases long-term security.

3.2. *Secure Transaction Signing Using Threshold Signatures*

The Threshold signatures enable numerous participants to engage in signing transactions, which do not disclose any private keys, safeguard authorization, avoid misuse, and decentralization of control over the process of validating the transactions.

$$h_m = sqrt\left(\left(\left(H(m)\right)^2 \right)^2 + \int_0^1 ((H(m) * t + t^3)^2)\, dt \right) \tag{14}$$

New message computation of the hash in the form of the message in the equation (14) converts the input message to a secure digest that has integrity and is resistant to alteration. The formulation underlines the non-linearity and diffusion with the higher-order and integral terms.

$$R = sqrt\left(\left(\left(\Sigma_{\{i=1\}_i^{\{t\}r}} \right)^2 \right)^2 + \int_0^1 \left(\left(\left(\Sigma_{\{i=1\}_i^{\{t\}r}} \right) * t + t^3 \right)^2 \right) dt \right) \tag{15}$$

In the case of collaboration signing, the random nonce aggregation in the equation below (15) is used to combine nonces of nodes participating in the signing process, making them unpredictable and eliminating the replay, or forgery, attacks.

$$\sigma = sqrt\left(\left(\left(\Sigma_{\{i=1\}_i^{\{t\}\sigma}} \right)^2 \right)^2 + \int_0^1 \left(\left(\left(\Sigma_{\{i=1\}_i^{\{t\}\sigma}} \right) * t + t^3 \right)^2 \right) dt \right) \tag{16}$$

The signature aggregation in Eq. (16) combines partial signatures created by several participants into a complete signature without correcting the signature but keeping the secrets of each key share secret.

$$V_{final} = sqrt\left(\left(\left(g^{\{\sigma\}mod} p - y^{\{h_m\}mod} p \right)^2 \right)^2 + \int_0^1 \left(\left(\left(\left(g^{\{\sigma\}mod} p - y^{\{h_m\}mod} p \right) * t + t^2 \right)^2 \right) \right) dt \right) \tag{17}$$

Lastly, using the aggregated signature in Eq. (17) validates it with the help of public parameters, which guarantees authenticity, accurately, and adversarial manipulation resistance in distributed settings.

4. Threshold Signature Generation Equations

The generation of threshold signature includes secret sharing, interpolation of polynomials, computation of partial signatures, and aggregation equations, which guarantee that they are correct, unforgeable, and resistant to adversarial reconstruction or key leakage.

$$S(x) = sqrt\left(\left(\left(k + \Sigma^i_{\{i=1\}_i^{\{t-1\}b}x} \right)^2 \right)^2 + \int_0^1 \left(\left(\left(k + \Sigma^i_{\{i=1\}_i^{\{t-1\}b}x} \right) * t + t^3 \right)^2 \right) dt \right) \tag{18}$$

Equation. (18) secret sharing initialisation builds a polynomial with the constant term as the secret, and the other coefficients of higher order, which create randomness to be distributed securely.

$$S_i = sqrt\left(\left(\left(S(i)\right)^2\right)^2 + \int_0^1 ((S(i) * t + t^2)^2)\, dt \right), i \in \{1, \dots, n\} \quad (19)$$

The sharing of the shares in the equation (19) calculations of the value of the polynomial at various indices are computed and then each member is given different share without the knowledge of the complete secret.

$$\sigma_i = sqrt\left(\left(\left(S_i * H(m)\right)^2\right)^2 + \int_0^1 \left(\left(\left(S_i * H(m)\right) * t + t^3\right)^2\right) dt \right) \quad (20)$$

The privacy of the individual parts remains a secret since the computation of partial signatures in the form of the equation (20) with a mixture of individual shares and the hash of the message to create secure signature fragments conceals the privacy of the individual parts.

$$\sigma_T = sqrt\left(\left(\left(\Sigma_{\{i=1\}_i^{\{t\}\lambda}} * \sigma_i \right)^2\right)^2 + \int_0^1 \left(\left(\left(\Sigma_{\{i=1\}_i^{\{t\}\lambda}\sigma_i} \right) * t + t^3 \right)^2 \right) dt \right) \quad (21)$$

Finally, the signature combination of (21) is a mixture of signatures with interpolation coefficients that are weighted, which produces a valid global signature without the loss of threshold security and correctness.

5. Result and Discussion

Results indicate improved security, reduced key compromise and fault tolerance and discourses are scalable with trade-offs in computational overhead and performance as compared to traditional single key cryptographic schemes.

Table 1

Comparative Performance Analysis of Traditional Single-Key

Metric	Traditional Single-Key System	Threshold Cryptography System
Security Strength (%)	85.2	96.7
Key Compromise Probability (%)	12.5	3.1
Fault Tolerance (Nodes %)	40	75
Transaction Signing Latency (ms)	8.4	11.2
Throughput (Transactions/sec)	520	485

As indicated in Table 1, threshold cryptography exhibits a significant security strength (96.7) compared to their conventional single-key systems (85.2%), and a large decrease in key compromise probability. Figure 2 compares

the traditional and threshold cryptography systems in terms of the key performance indicators.

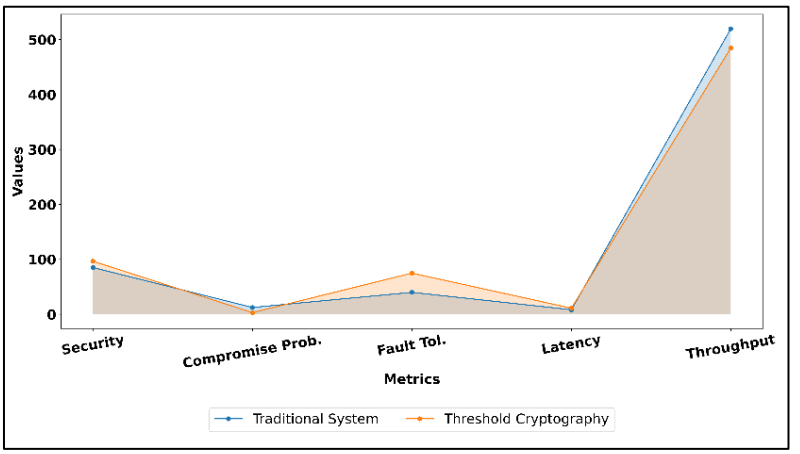


Figure 2

Comparative Analysis of Traditional and Threshold Cryptography Systems

It has nearly twice the fault tolerance, implying that it is more robust in a distributed environment. Figure 3 shows stacked visualization of a comparison of some of the metrics of the cryptography system performance.

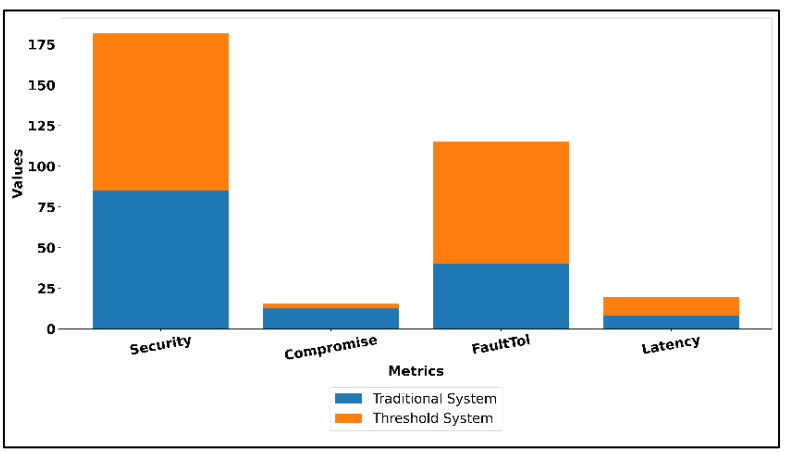


Figure 3

Stacked Performance Visualization of Cryptographic System Metrics

These advantages are associated with a moderate increase in the transaction signing time, and a minor reduction in throughput. Overall, the trade-

off between enhanced security, reliability and decentralized trust and relatively small performance overheads is biased towards the former.

6. Conclusion

Threshold cryptography can be used to provide a secure and scalable method in addressing crucial security concerns in distributed ledger systems. Decentralizing the control of the private keys to the numerous participants is beneficial in eliminating the single points of failure, and significantly reducing the risks associated with key compromise and insider threats. The blockchain architectures will enhance trust, fault tolerance and decentralization at the point of embarkation of the blockchain schemes in the cryptography of transactions and wallet management. The approach is also correct and robust in adversarial conditions due to mathematical modeling of threshold signature generation. Although this involves a slight computational cost and complexity in coordination, in practice in most realistic scenarios, the security benefits outweigh these limitations.

References

[1] A. A. Khan and L. Y. Por, "Special Issue on Information Security and Cryptography: The Role of Advanced Digital Technology," *Applied Sciences*, vol. 14, no. 5, Art. no. 2045 (2024), doi: 10.3390/app14052045.

[2] B. Schoenmakers and T. Segers, "Secure Groups for Threshold Cryptography and Number-Theoretic Multiparty Computation," *Cryptography*, vol. 7, no. 4, Art. no. 56 (2023), doi: 10.3390/cryptography7040056.

[3] J. Ni, G. Fang, Y. Zhao, J. Ren, L. Chen, and Y. Ren, "Distributed Group Key Management Based on Blockchain," *Electronics*, vol. 13, no. 11, Art. no. 2216 (2024), doi: 10.3390/electronics13112216.

[4] P. Khobragade, P. K. Dhankar, P. K. Adakane, M. Dhone, S. A. Thakur, and P. Saraf, "Blockchain-Based Solutions for Enhancing Data Security in Cloud Computing Environments," in *Proc. 2025 1st Int. Conf. Data Sci. Intell. Netw. Comput. (ICDSINC)*, Raipur, India, pp. 599–604 (2025), doi: 10.1109/ICDSINC66221.2025.11448121.

[5] D. Romano and G. Schmid, "Beyond Bitcoin: Recent Trends and Perspectives in Distributed Ledger Technology," *Cryptography*, vol. 5, no. 4, Art. no. 36 (2021), doi: 10.3390/cryptography5040036.

[6] J. Shi, X. Zeng, and R. Han, "A Blockchain-Based Decentralized Public Key Infrastructure for Information-Centric Networks," *Information*, vol. 13, no. 5, Art. no. 264 (2022), doi: 10.3390/info13050264.

[7] K. A. Waghmare, P. Mahalle, S. N. Ajani, K. Gupta, S. S. Badhiye, and P. Shende, "Integration of blockchain technology with cryptographic solutions for robust healthcare information security," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 28, no. 5-A, pp. 1589–1598 (2025), doi: 10.47974/JDMSC-2157.

[8] Z. A. Shaikh, A. A. Khan, L. Baitenova, G. Zambinova, N. Yegina, N. Ivolgina, A. A. Laghari, and S. E. Barykin, "Blockchain Hyperledger with Non-Linear Machine Learning: A Novel and Secure Educational Accreditation Registration and Distributed Ledger Preservation Architecture," *Applied Sciences*, vol. 12, no. 5, Art. no. 2534 (2022), doi: 10.3390/app12052534.

[9] R. Shevchuk, I. Lishchynskyy, M. Ciura, M. Lyzun, R. Kozak, and M. Kasianchuk, "Application of Blockchain Technology in Emergency Management Systems: A Bibliometric Analysis," *Applied Sciences*, vol. 15, no. 10, Art. no. 5405 (2025), doi: 10.3390/app15105405.

[10] S. K. Mandal, R. Singh, N. Chandu, D. Mehta, S. K. Henge, D. Kothapeta, C. K. Hinge, A. Sharma, and T. Hussain, "Evolutionary multi-authentication cryptographic key implications for secure data access control," *Journal of Discrete Mathematical Sciences and Cryptography*, vol. 28, no. 5-B, pp. 1865–1874 (2025), doi: 10.47974/JDMSC-2363.

[11] C. D. Morar and D. E. Popescu, "A Survey of Blockchain Applicability, Challenges, and Key Threats," *Computers*, vol. 13, no. 9, Art. no. 223 (2024), doi: 10.3390/computers13090223.

[12] R. Saia, A. S. Podda, L. Pompianu, D. Reforgiato Recupero, and G. Fenu, "A Blockchain-Based Distributed Paradigm to Secure Localization Services," *Sensors*, vol. 21, no. 20, Art. no. 6814 (2021), doi: 10.3390/s21206814.

[13] P. A. D. S. N. Wijesekara and S. Gunawardena, "A Review of Blockchain Technology in Knowledge-Defined Networking, Its Application, Benefits, and Challenges," *Network*, vol. 3, no. 3, pp. 343–421 (2023), doi: 10.3390/network3030017.

[14] R. C. Poonia, T. Singh, C. K. Rajashri, D. Mondal, and S. M. Patil, "Optimizing communication system security through comprehensive testing and intelligent network configurations," *Journal of Information and Optimization Sciences*, vol. 47, no. 5-A, pp. 1709–1717 (2026), doi: 10.47974/JIOS-2261.

Received April, 2026