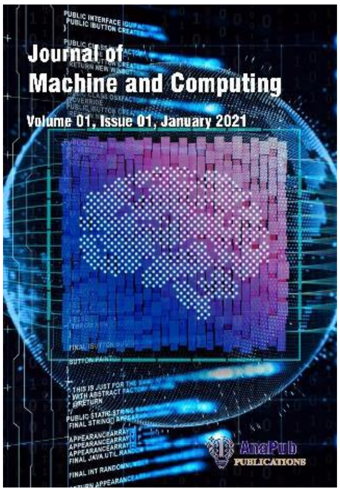
Journal Pre-proof

Mitigating Data Tampering in Smart Grids Through Community Blockchain-Driven Traceability Frameworks



Gayathri Ananthakrishnan, Sudhakar Sengan, Mohanraj E, Thirumoorthy Palanisamy, Veeramallu B and Srinivasarao B

DOI: 10.53759/7669/jmc202505138

Reference: JMC202505138

Journal: Journal of Machine and Computing.

Received 18 February 2025

Revised from 26 May 2025

Accepted 16 June 2025

Please cite this article as: Gayathri Ananthakrishnan, Sudhakar Sengan, Mohanraj E, Thirumoorthy Palanisamy, Veeramallu B and Srinivasarao B, "Mitigating Data Tampering in Smart Grids Through Community Blockchain-Driven Traceability Frameworks", Journal of Machine and Computing. (2025). Doi: https:// doi.org/10.53759/7669/jmc202505138.

Mitigating Data Tampering in Smart Grids Through Community Blockchain-Driven Traceability Frameworks

Gayathri Ananthakrishnan[1,*], Sudhakar Sengan[2], E. Mohanraj[3], Thirumoorthy Palanisamy[4], B. Veeramallu[5], B. Srinivasarao[6]

[1]Department of Information Technology, School of Computer Science Engineering and Information Systems, Vellore Institute of Technology, Vellore, 632014, Tamil Nadu, India.

**Corresponding Author Email : gayathri.a@vit.ac.in*

[2]Department of Computer Science and Engineering, PSN College of Engineering and Technology, Tirunelveli, 627451, Tamil Nadu, India. *Email: sudhasengan@gmail.com*

[3]School of Computing, SRM Institute of Science and Technology, Tiruchirappalli, Tamil Nadu, 621105, India. *Email : csemohanraj@gmail.com*

[4]Department of Computer Science and Engineering, Erode Sengunthar Engineering College, Thuduppathi, Erode, Tamil Nadu 638057, India. *Email : thiru4u@gmail.com*

[5]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur, Andhra Pradesh, 522302, India. *Email: bvmallu@kluniversity.in*

[6]Department of Computer Science and Engineering Lakireddy BaliReddy College of Engineering, Mylavaram, NTR Dt, Andhra Pradesh, India. *Email: dr[illegible]2@gmail.com*

Abstract

Data integrity in Smart Grids (SG) systems can be vulnerable with the implementation of the novel Community Blockchain-Driven Traceability Framework (CBDTF). It enhances Detection Rates (DR), maintains low End-to-End Delay (EED), and uses less energy by using distributed ledger technology and community-based validation. This model deployed a Delegated Proof of Stake (DPoS) consensus mechanism and community-driven testing, resulting in an average Detection Rate (DR) of 98.7% for Data Tampering attacks and a False Positive Rate (FPR) of 1.78%. It outperforms conventional Blockchain (BC) solutions with an EED of 120.8 ms and an average CPU utilization of 1,113 *tx/kWh*. When compared with conventional Proof-of-Work (PoW), CBDTF requires 60% less energy while proving 96.2% consensus resilience against distinct attacks. Applying real-world SG data collected by a distributed network of 100 nodes, the accuracy of this model was tested. The present study makes a valuable contribution to the field by signifying how BC platforms driven by the public can address SG's data security issues while maintaining the accuracy of real-time operations.

Keywords: Smart Grids, Data Tampering, Blockchain, Data Integrity, Attacks, Security.

1. Introduction

The Smart Grid (SG) has revolutionized power systems, transforming traditional power systems into advanced sensing, communication, and control technologies [1-3]. This has led to increased vulnerabilities to Data Tampering (DT) and cyber-attacks, compromising grid stability, incorrect hyping, and potentially causing network failures [4-5]. Traditional security mechanisms challenge the distributed nature of current SG and the requirement for real-time data validation [6-8]. The SG has improved grid monitoring, demand response, and the efficient integration of Distributed Energy Resources (DER). However, it has also expanded the attack surface for malicious actors due to the complex network of interconnected devices, creating multiple entry points for data manipulation [9].

The integrity of data in SG is of primary importance for several reasons.

a) Operational decisions heavily rely on the accuracy of tests from grid components, such as smart meters, Phasor Measurement Units (PMUs), and Supervisory Control and Data Acquisition (SCADA) systems.
b) Financial transactions and billing processes are reliant on reliable consumption data [11].
c) The practical operation of grid stability and security mechanisms demands the use of reliable real-time data.

Data integrity in grid operations can lead to financial losses and disruptions [12]. Current security solutions in SG face limitations, including single points of failure, scalability challenges, and limitations in traditional cryptographic methods. Blockchain Technology (BT)-based solutions also introduce End-to-End Delay (EED) and energy overhead, making them unsuitable for real-time grid operations [13-[illegible]].

The proposed Community Blockchain-Driven Traceability Framework (CBDTF) addresses Energy Efficiency (EE) and transaction Network Throughput (NT) limitations in conventional BT implementations by leveraging community participation and specialized consensus mechanisms designed explicitly for SG, thereby enhancing BT's potential.

This paper presents several key contributions:

1) A new multi-layered BT was explicitly developed for SG data validation.
2) A consensus mechanism that is energy-efficient and secure without compromising performance despite its EE.
3) A technique of validation that is determined by the community and improves attack Detection Rate (DR) while also reducing the probability of False Positives Rates (FPR).
4) The development of a robust traceability model that enables real-time auditing and verification of grid data.
5) The use of real-world SG data and attacks for substantial test validation

The remainder of this paper is organized as follows: Section 2 reviews relevant literature and identifies current research gaps. Section 3 presents the proposed model, technical methodology, and implementation details. Section 4 outlines the experimental setup and evaluation metrics. Section 5 presents results and discussion, including comparison with baseline approaches. Finally, Section 6 concludes the paper and recommends future research directions.

2. Literature Survey

This survey examines recent research and developments in the deployment of BT in Singapore, focusing on its key role in addressing security challenges in the sector to enhance transparency and efficiency.

The authors [15] propose a BT-based security model for the Smart Grid (SG) that focuses on secure authentication and efficient data sharing among distributed devices. They introduce redesigned blocks and gateway nodes for device identity verification and implement a multi-layer Smart Contract (SC) for secure interactions. The IEEE Smart Grid Bulletin discusses BT's potential to address cybersecurity issues in the SG but notes challenges such as scalability and the requirement for standardized consensus algorithms.

Recent advancements have explored the combination of BT with Wireless Sensor Networks (WSN) to secure SG data [15], thereby ensuring data integrity and authenticity. RETINA, a model that utilizes BC for distributed and secure trust management in SG applications, integrates Public Key Infrastructure (PKI) and Web of Trust (WoT) concepts to facilitate decentralized communication and robust key management. It also incorporates an SC-based energy trading mechanism to promote the use of Renewable Energy (RE), taking into account factors such as trust and energy type.

The study [16] proposes an incentive mechanism for BT-based data sharing among multiple operators in Singapore to combat False Data Injection (FDI) attacks, ensuring data integrity and enhancing grid security by penalizing anomalies.

The survey [17] explores the integration of BC and SG in energy management, security, and privacy control, addressing challenges such as low processing NT and privacy issues, and provides insights for future research.

Comparably, [18] performed a detailed examination of BT applications in the energy sector, identifying possibilities and challenges in the method of implementing BT for the aim of enhancing the security and efficiency of SG.

3. Proposed Methodology

3.1. Model of the CBDTF

The CBDTF uses a multi-layered model (Figure 1) that integrates SG setup with Distributed Ledger Technology (DLT) and encourages community participation for enhanced security and transparency. This architecture, denoted as system Ψ, consists of four interconnected layers:

the Data Collection Layer (DCL), the Blockchain Integration Layer (BIL), the Community Consensus Layer (CCL), and the Traceability Management Layer (TML). Together, these layers ensure robust data integrity, traceability, and resilience against DT.

i. **DCL:** The DCL serves as the primary interface with the SG's data collection components. The SG setup includes data sources $S = \{s_1, s_2, \ldots, s_n\}$, which continuously generates raw time-series data points $'d_i(t)'$. These raw sizes, $X(t) = \{x_1(t), x_2(t), \ldots, x_m(t)\}$, experience a preprocessing function $\phi(x_i(t))$ to standardize, clean, and validate the data.

The preprocessing includes data standardization and preliminary validation, Eq. (1).

$$\phi(x_i(t)) = \begin{cases} x_i'(t), & \text{If } V(x_i(t)) = \text{ True} \\ \text{NULL}, & \text{Otherwise} \end{cases} \quad (1)$$

Where,

- $x_i'(t)$ → The validated measurement
- $V(x_i(t))$ i→ A validation function ensuring adherence to predefined consistency checks.

ii. **BIL:** The BIL is responsible for generating and maintaining the BT. Validated data points $'x_i'(t)'$ are grouped into blocks $B = \{b_1, b_2, \ldots, b_k\}$. Each block $'b_i'$ comprises several vital components:

- **Timestamp** $(\boldsymbol{\tau_i})$ **:** Records the creation time of the block.
- **Previous Block Hash** $(\boldsymbol{h(b_{i-1})})$**:** Ensures block immutability and order.
- **Merkle Root (MR) as** $(\boldsymbol{M_r(T)})$**:** The root hash of all transactions $'T'$ within the block.
- **Validator Signatures** $(\boldsymbol{\Sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_j\}})$ **:** Captures approvals from community validators.

The function defines the block creation process, as shown in Eq. (2).

$$\beta\left(x_i'(t), \tau_i, h(b_{i-1})\right) \to b_i \quad (2)$$

Where,

β → The validated data, timestamp

- The hash of the previous block to generate the new block $'b_i'$.

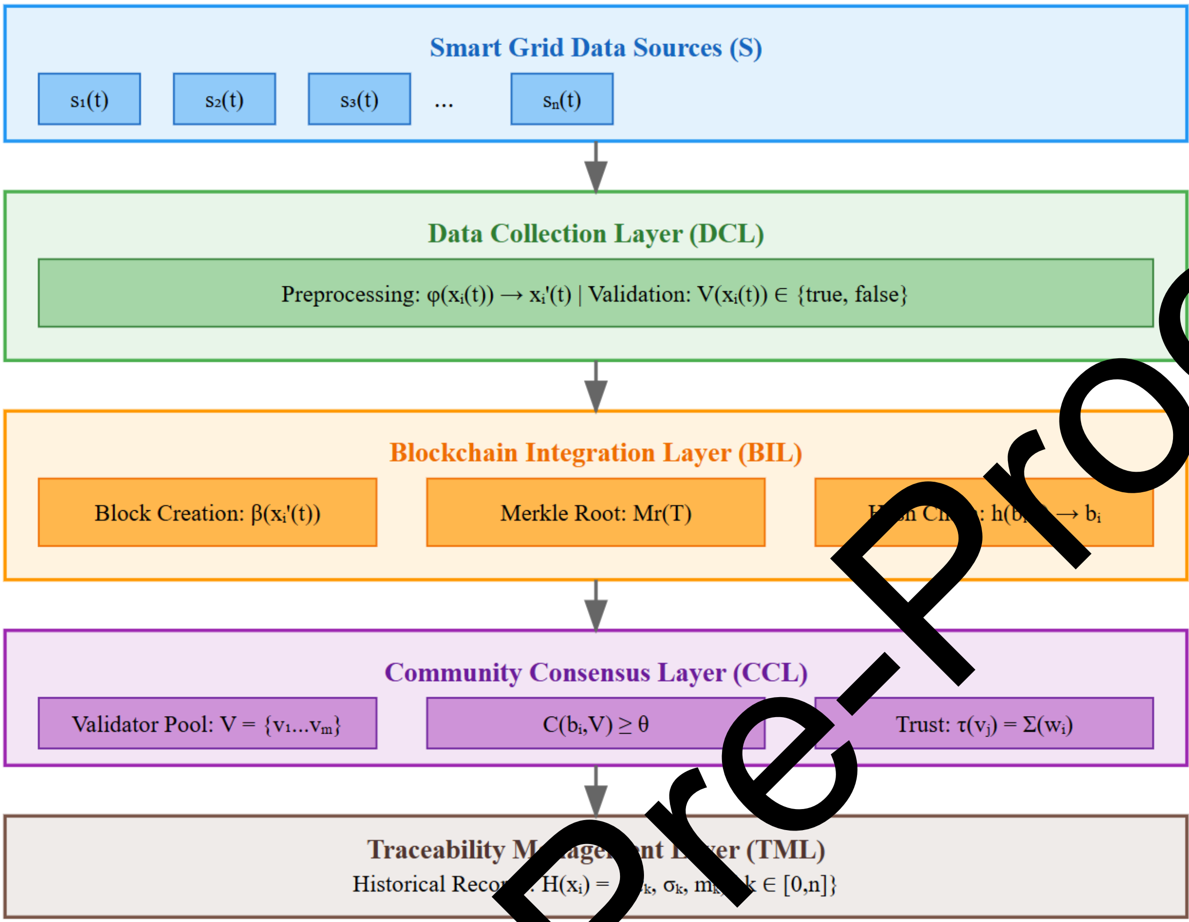


Figure 1: Proposed CBDTF Model

iii. **CCL:** The CCL implements a consensus mechanism $'C'$ to validate and approve new blocks. A community pool $P = \{p_1, p_2, \dots, p_m\}$ provides a set of validators $V = \{v_1, v_2, \dots, v_m\}$. Each validator evaluates the integrity of the block $'b_i'$ and casts a weighted vote based on their trust score $'w_j'$. Consensus is achieved if the weighted sum of agreements surpasses a threshold $'\theta'$, Eq. (3).

$$\sum_{j=1}^{m} \left(w_j \cdot a_j(b_i)\right) \geq \theta \quad (3)$$

Where,

- w_j → Dynamically adjusted based on each validator's historical reliability, responsiveness, and peer evaluations.

iv. **TML:** The TML maintains a comprehensive historical record $'H(x_i)'$ of all data points and their associated metadata. For each data point $'x_i(t)'$, the traceability function $'T'$ maps it to its historical record:

$$H(x_i) = \{(\tau_k, \sigma_k, m_k) \mid k \in [0, n]\}, \quad (4)$$

Where,

- τ_k → The timestamp

- σ_k → The validator's signature
- m_k → The metadata.

The layer ensures any modification $\mu(x_i(t))$ to a data point is immutably recorded and verifiable, Eq. (5).

$$\forall \mu(x_i(t)), \exists \sigma_j \in \Sigma: \text{Verify}\left(\sigma_j, \mu(x_i(t))\right) = \text{True} \tag{5}$$

3.2 SG Data Collection and Preprocessing

The SG data collection and preprocessing phase forms the first layer of the proposed context, ensuring that raw data from diverse sources is prepared for secure and efficient integration into the BT. The process involves structured data acquisition, validation, cleaning, and transformation to maintain accuracy, consistency, and reliability.

1 **Data Collection from SG Devices:** The SG set-up comprises numerous devices, $S = \{s_1, s_2, \dots, s_n\}$, such as smart meters, sensors, and actuators, which continuously generate time-series data.

The measurements collected at time $'t'$ are represented as Eq. (6).

$$\mathrm{X}(t) = \{x_1(t), x_2(t), \dots, x_m(t)\} \tag{6}$$

Where,

- $x_i(t)$ → The raw size from the i^{th} device.
- $'m'$ → The sum of measurements.
- $x_i(t)$ → Connected with metadata, including device ID, timestamp, and location, as $\mathrm{M}_i(t) = \{\mathrm{ID}, \tau, \mathrm{Loc}\}$.

2 **Data Validation and Standardization:** To ensure the integrity and usability of the data, a preprocessing function $\phi(x_i(t))$ is applied, encompassing validation and standardization steps, Eq. (7).

$$\phi(x_i(t)) = \begin{cases} x_i'(t), & \text{If } V(x_i(t)) = \text{True} \\ \text{Null}, & \text{Otherwise} \end{cases} \tag{7}$$

Where,

- $V(x_i(t))$ → Validation function that checks each data point for anomalies such as missing values, outliers, or invalid formats. If $V(x_i(t))$ evaluates to True, the measurement $x_i(t)$ is transformed into a validated and standardized form $x_i'(t)$; otherwise, it is discarded. Validation involves threshold checks and outlier DR using statistical methods.

For instance, if the expected range for a measurement $x_i(t)$ is $[a, b]$, the validation is expressed as Eq. (8)

$$V(x_i(t)) = \begin{cases} \text{True}, & \text{If } a \le x_i(t) \le b \\ \text{False}, & \text{Otherwise} \end{cases} \tag{8}$$

3 **Handling Missing Data:** In cases where measurements contain missing values, imputation methods are employed. Let $\mathrm{X}_{\text{Missing}}(t) \subset \mathrm{X}(t)$ as the set of missing data points. These are replaced using predictive imputation methods (IMP), such as linear interpolation or Machine Learning (ML)-based predictions, as shown in Eq. (9).

$$x_i(t) = \text{IMP}\,(\mathrm{X}_{\text{Context}}) \tag{9}$$

Where,

- $\mathrm{X}_{\text{Context}}$ → The contextual data surrounding $\mathrm{x_i(t)}$.

4 **Data Transformation and Normalization:** After validation, the data points are normalized to ensure compatibility across different devices and metrics. Let $x_i'(t)$ represent the validated measurement.

The normalization function $N(x_i'(t))$ transforms the data into a standardized range, *e.g.,* $[0, 1]$, using Eq. (10).

$$N(x_i'(t)) = \frac{x_i'(t) - \mathrm{Min}(\mathrm{X}')}{\mathrm{Max}(\mathrm{X}') - \mathrm{Min}(\mathrm{X}')}, \tag{10}$$

Where,

- $\mathrm{Max}(\mathrm{X}')$, $\mathrm{Min}(\mathrm{X}')$ → The maximum and minimum values in the validated dataset.

5 **Temporal Alignment:** The SG devices frequently generate data at varying intervals. To maintain temporal consistency, all measurements are resampled to a standard time interval Δt.

The resampling function $R(\cdot)$ ensures uniform timestamps, Eq. (11)

$$\mathrm{X}'_{\text{aligned}}(t) = R(\mathrm{X}'(t), \Delta t) \tag{11}$$

Where,

- $\mathrm{X}'_{\text{Aligned}}(t)$ → The temporally aligned dataset.

Given the raw data $\mathrm{X}(t)$, the final preprocessed dataset $\mathrm{X}_{\text{Final}}(t)$ is computed as, Eq. (12)

$$\mathrm{X}_{\text{Final}}(t) = \{N(\phi(x_i(t))) \mid V(x_i(t)) = \text{True}, \forall i\} \tag{12}$$

This pre-processed dataset is then forwarded to the BT integration layer for secure storage and further analysis.

3.3 Blockchain Integration Mechanism (BIM)

The BIM (Figure 2) is a pivotal component of the proposed model, designed to securely manage SG data by organizing, validating, and storing it in a decentralized and immutable ledger. The mechanism converts pre-processed data into secure BT transactions, ensuring consensus and synchronization across a distributed network using transaction development, block creation, cryptographic linkage, decentralized consensus, and ledger synchronization.

The process begins with the transformation of validated data points as $\mathrm{X}_{\text{final}}(t) = \{x_1'(t), x_2'(t), \ldots, x_m'(t)\}$, into BT-compatible transactions. Each transaction $T_i(t)$ encapsulates a data payload $x_i'(t)$, metadata $M_i(t)$ including device ID, timestamp, and location, and a unique transaction identifier TxID_i. The identifier is generated using a cryptographic hash function $H(\cdot)$, ensuring the uniqueness and integrity of the transaction, Eq. (13)

$$\mathrm{TxID}_i = H(x_i'(t) \parallel M_i(t)) \tag{13}$$

Where,

- $\parallel \rightarrow$ concatenation.
- These transactions form a transaction set $\mathrm{T}(t) = \{T_1(t), T_2(t), \ldots, T_m(t)\}$, which serves as the primary input for block creation. The validated transactions are grouped into blocks, represented as $'B_k'$, where $'k'$ denotes the block index. Each block contains two main components: a header and a body. The header includes critical elements such as the block index $'k'$, a timestamp $'\tau_k'$, the cryptographic hash of the previous block $h(B_{k-1})$, and an MR as $M_r(\mathrm{T})$.

By iteratively hashing pairs of transactions to generate a root hash, the MR securely encapsulates all block transactions, as shown in Eq. (14).

$$M_r(\mathrm{T}) = H\big(H(T_1) \parallel H(T_2)\big) \parallel H\big(H(T_3) \parallel H(T_4)\big) \ldots \tag{14}$$

This structure ensures the integrity and traceability of individual transactions, as any modification to a transaction will result in a mismatch of the MR, thereby invalidating the block. The block's body contains the transaction set $'\mathrm{T}(t)'$, providing the complete list of validated transactions stored in the block.

To ensure the block's immutability, a cryptographic hash $h(B_k)$ is computed for the entire block, encompassing its header and body, Eq. (15).

$$h(B_k) = H(\tau_k \parallel h(B_{k-1}) \parallel M_r(\mathrm{T}) \parallel \mathrm{T}(t)) \tag{15}$$

This hash uniquely identifies the block and links it to its predecessor in the BT, establishing a secure and tamper-proof chain.

To validate and add a block to the BC, a decentralized consensus mechanism $'C'$ is employed, leveraging the participatory role of validators $V = \{v_1, v_2, \ldots, v_n\}$. The BC protocol enables validators to independently assess the integrity and validity of the block. The consensus process aggregates validator votes, weighted by trust scores of 'w_i', to determine block approval, which is accepted if the weighted sum meets a predefined threshold of 'θ', Eq. (16).

$$C(B_k, V) = \text{ True } \Leftrightarrow \sum_{i=1}^{n} \left(w_i \cdot v_i(B_k)\right) \geq \theta \quad (16)$$

This decentralized validation prevents any single entity from DT with the ledger, thereby protecting its integrity and availability.

Once consensus is achieved, the validated block is appended to the global ledger $\mathcal{L} = \{B_1, B_2, \ldots, B_k\}$. All nodes in the BT network synchronize their copies of $'\mathcal{L}'$ to ensure consistency. This synchronization is verified through a ledger consistency function $'\xi(\mathcal{L})'$, which compares the block hashes across all nodes, Eq. (17).

$$\xi(\mathcal{L}) = \text{ True } \Leftrightarrow h(B_k) \text{ Matches across all nodes.} \quad (17)$$

The BIM's security is based on cryptographic basics and decentralized networks. The cryptographic linkage between blocks prevents unauthorized modifications, while the decentralized consensus mechanism distributes control among multiple validators. The immutable ledger maintains a transparent record of all SG transactions, enhancing accountability and trust in the system (Figure 2).

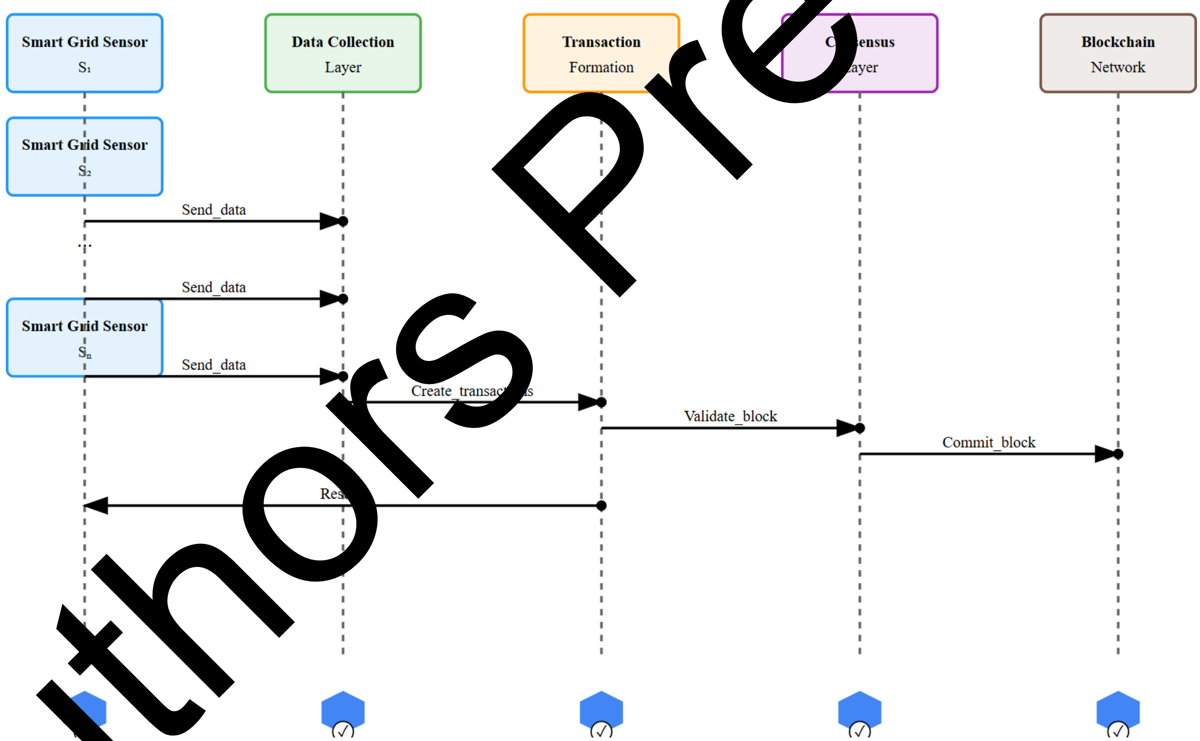


Figure 2: Overall process of BIM

3.4 Traceability and Data Verification Protocols

The proposed model includes Traceability and Data Verification Protocols (Figure 3), which ensure transparent auditability, cryptographic security, and verifiability of all data in the SG ecosystem. These protocols combine cryptographic principles, blockchain immutability, and community-driven consensus mechanisms to ensure robust data integrity.

1 **Data Traceability Model:** The traceability protocol sets an unbroken chain of provenance for every data point $'x_i'(t)'$ within the SG. The historical lineage of a data point is captured as Eq. (18)

$$T(x_i'(t)) = \{(t_k, \sigma_k, \mu_k) \mid k \in [0, n]\} \tag{18}$$

Where:

- t_k→ The timestamp of a specific operation (generation, validation, or modification) on $x_i'(t)$,
- σ_k→The cryptographic signature of the validator that authorized the operation, and
- μ_k→ Operation metadata, including the type of action and associated parameters.

The traceability mechanism uses BT's inherent immutability to ensure all transactions involving $x_i'(t)$ are recorded in linked blocks. Each transaction $'T_i It\ is\ cryptographically\ hashed\ as\ shown\ in$ Eq. (19).

$$h(T_i) = H(x_i'(t) \parallel t \parallel M_i) \tag{19}$$

Where,

- M_i→ Metadata such as device ID and geographic location.

These hashed transactions are organized within a block $'B_k'$, linked by the MR as M_r(T), Eq. (20).

$$M_r(\mathrm{T}) = H\big(H(T_1) \parallel H(T_2)\big) \parallel \cdots \tag{20}$$

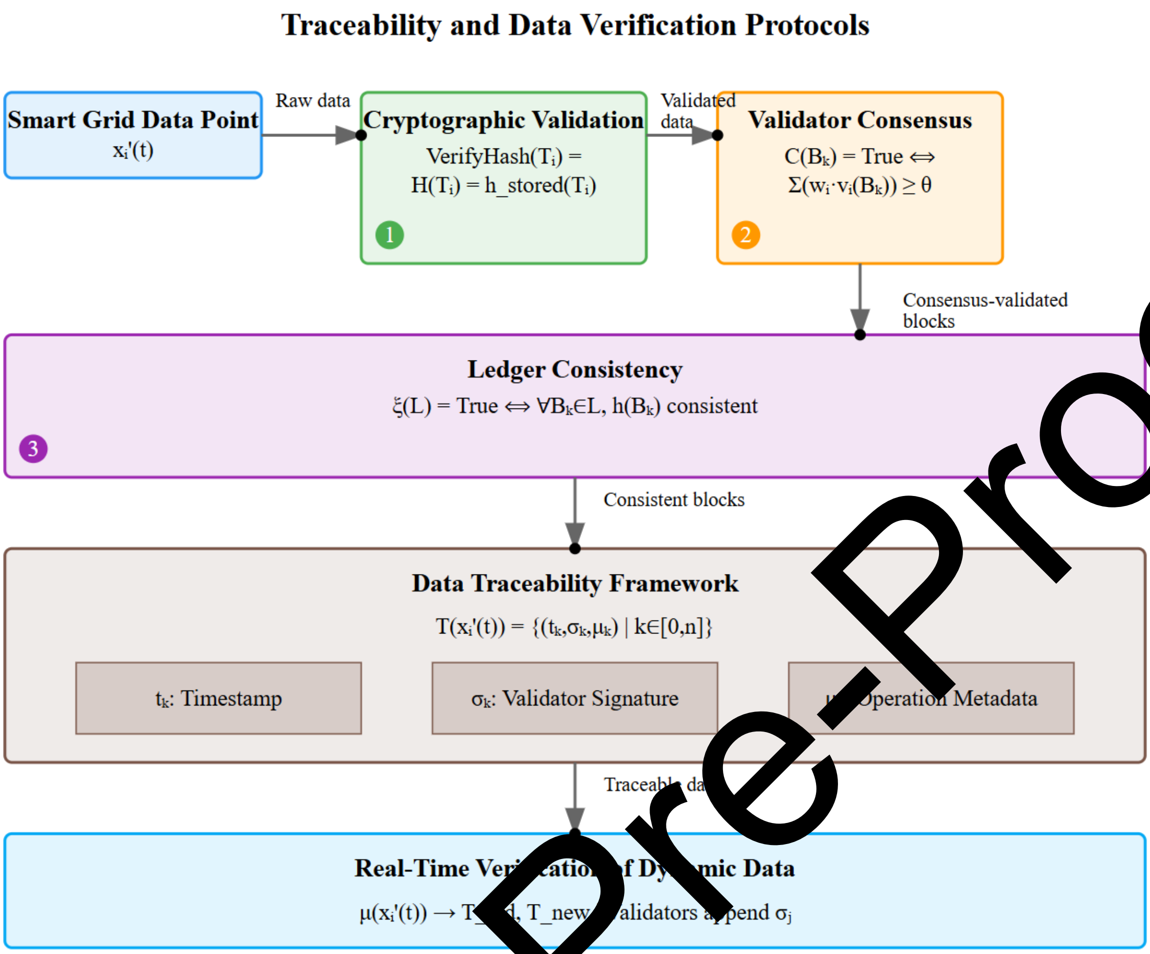


Figure 3: Traceability and Data Verification

The model ensures instant detection of any modification to 'T_i' due to a mismatch in the MR, allowing stakeholders to reconstruct the entire operational history of a data point. A query to the BC as $h(x_i'(t))$ retrieves all associated transactions $\{T_i\}$, providing a verifiable record of changes.

2 **Data Verification Protocols:** In BC, data verification protocols ensure data authenticity, integrity, and network synchronization using cryptographic validation, validator consensus, and ledger consistency at three primary levels.

- **Cryptographic Validation:** Cryptographic validation guarantees that the content of each transaction has not been altered. For any transaction $'T_i'$ containing $x_i'(t)$, Its integrity is verified by recalculating the hash and comparing it with the stored hash, as shown in Eq. (21).

$$\text{VerifyHash}\ (T_i) = \begin{cases} \text{True}, & \text{if } H(T_i) = h_{\text{stored}}\ (T_i), \\ \text{False}, & \text{otherwise.} \end{cases} \tag{21}$$

This step ensures that even a minor alteration to T_i or $x_i'(t)$ renders the transaction invalid.

- **Validator Consensus:** Each block $'B_k'$ experiences a decentralized consensus process before being attached to the BC. Validators $V = \{v_1, v_2, \ldots, v_n\}$, selected from the community, independently verify the block's compliance with protocol rules.

The consensus decision is formalized as Eq. (22)

$$C(B_k) = \text{ True } \Leftrightarrow \sum_{i=1}^{n} \left(w_i \cdot v_i(B_k)\right) \geq \theta \tag{22}$$

Where:

- $w_i \rightarrow$ The trust score of the validator v_i,
- $v_i(B_k) \rightarrow$ The validator's vote (1 for approval, 0 for rejection),
- $\theta \rightarrow$ The predefined consensus threshold.

Data validation is decentralized, reducing the risk of centralized attacks.

- **Ledger Consistency:** To maintain synchronization across the distributed ledger $'\mathcal{L}'$, each node periodically validates the integrity of its BC copy. This is achieved using a ledger consistency function $'\xi(\mathcal{L})'$, which compares the hashes of all blocks, Eq. (23).

$$\xi(\mathcal{L}) = \text{ True } \Leftrightarrow \forall B_k \in \mathcal{L}, h(B_k) \text{ is consistent across nodes.} \tag{23}$$

Inconsistencies trigger a reconciliation protocol to restore uniformity, preserving the blockchain's reliability.

3 **Real-Time Verification of Dynamic Data:** The model supports real-time data verification, addressing scenarios where data points are dynamically updated in real-time.

Each modification $\mu(x_i'(t))$ results in a new transaction T_{new} while preserving the original transaction T_{old} for auditability, Eq. (24).

$$\mu(x_i'(t)) \rightarrow T_{\text{old}}, T_{\text{new}} \tag{24}$$

Where:

- Validators review $'T_{\text{New}}'$ and append their cryptographic signatures σ_j, ensuring that every modification is authorized and traceable. The BT maintains the current state of $x_i(t)$ and its historical record.

3.5 Cryptographic Techniques for Data Security

The proposed model uses cryptographic methods to ensure the integrity, authenticity, and confidentiality of SG data throughout its lifecycle. These methods utilize cryptographic hashing, digital signatures, and secure key management to establish a robust foundation for tamper-resistant and verifiable data storage. At the core of data security is cryptographic hashing, which ensures that any variation to data is directly measurable. Each validated data point $'x_i'(t)'$ is hashed using a cryptographic hash function $'H(\cdot)'$, producing a fixed-length digest, Eq. (25).

$$h(x_i'(t)) = H(x_i'(t)) \quad (25$$

Where,

- This hash is unique to $'x_i'(t)'$ and is computationally infeasible to reverse-engineer or replicate for different inputs, ensuring the integrity of the data. In the BT, hashed transactions are aggregated into a Merkle Tree (MT), with the MR as $'M_r'$ representing the combined integrity of all transactions in a block, Eq. (26).

$$M_r = H\big(H(T_1) \parallel H(T_2)\big) \parallel \cdots \quad (26)$$

Where,

- T_i → The hash of transaction $'i'$.
- If any transaction $'T_i'$ is altered, the change propagates by the MT, invalidating the block's cryptographic hash and breaking the BT's integrity.

The model uses digital signatures to ensure that all transactions and blocks are authorized. Each validator $'v_j'$ in the network is assigned a private key k_j^{Priv} for signing and a public key k_j^{Pub} for verification. A transaction $'T_i'$ is signed by a validator using their private key, Eq. (27).

$$\sigma_j = \text{Sign}\left(T_i, k_j^{\text{Priv}}\right) \quad (27)$$

- σ_j → The transaction, enabling network participants to verify the validator's authenticity, Eq. (28).

$$\text{Verify}\left(\sigma_j, T_i, k_j^{\text{Pub}}\right) = \begin{cases} \text{True}, & \text{If the signature is valid,} \\ \text{False}, & \text{Otherwise.} \end{cases} \quad (28)$$

Digital signatures and secure key management are employed in a system to ensure the traceability and trustworthiness of data while also maintaining the confidentiality of sensitive information through encryption and decryption. Public-key (PuK) cryptography helps secure key exchange between participants. Let $'K'$ → a symmetric key used for data encryption. The sender encrypts $'K'$ using the recipient's PuK as K_{Pub}, Eq. (29).

$$C_K = \text{Encrypt}\,(K, K_{\text{Pub}}) \quad (29)$$

The recipient decrypts C_K using their Private Key (PrK) as K_{priv}, Eq. (30).

$$K = \text{Decrypt}\,(C_K, K_{\text{Priv}}) \quad (30)$$

This ensures that the symmetric key remains secure even if the key exchange is intercepted, enabling encrypted data transmission.

3.6 Consensus Algorithm

The Delegated Proof of Stake (DPoS) was selected for the proposed BT due to its suitability for SG's unique requirements, including high transaction NT, low EED, EE, decentralization, and resilience against adversarial behavior, following an evaluation of various consensus protocols.

The exponential development of data generated by DER and Internet of Things (IoT) devices as $S = \{s_1, s_2, \dots, s_n\}$. An SG environment demands efficient BT consensus mechanisms to process this data effectively. Sustainability prioritizes sustainability, making energy-intensive mechanisms, such as Proof of Work (PoW), unsuitable. The dynamic and decentralized nature of SG demands a consensus algorithm that can adapt to network changes and provide robust fault tolerance to mitigate risks from malicious nodes or network disruptions.

In the DPoS, block validation is delegated to a predefined set of validators $V = \{v_1, v_2, \dots, v_m\}$, where $'m'$ is the total number of validators selected by stakeholder voting. Validators are responsible for proposing and validating blocks in a deterministic, round-robin manner, which significantly reduces competition and achieves predictable performance. Let $'B_k'$ represent the k^{th} block to be validated and $'w_i'$ the voting weight of the validator $'v_i'$, derived from the proportion of stakeholder votes received.

The decision to approve a block $'B_k'$ is governed by the weighted consensus function, Eq. (31).

$$C(B_k) = \text{ True } \Leftrightarrow \sum_{i=1}^{m} w_i \cdot v_i(B_k) \geq \theta, \tag{31}$$

Where:

- $v_i(B_k) \in \{0, 1\}$ i→ Validator v_i 's approval (1) or rejection (0) of B_k,
- θ → The consensus threshold, typically set as a supermajority ($\theta > 0.67$) to ensure robustness against adversarial actions.

Each validator $'v_i'$ is incentivized to act honestly through a staking mechanism, where their stake $'S_i'$ represents collateral that can be forfeited in the event of malicious activity. The probability of selecting a validator is proportional to their voting weight, Eq. (32).

$$P(v_i) = \frac{w_i}{\sum_{j=1}^{m} w_j} \tag{32}$$

This ensures that validators with higher trust and stake are more likely to contribute to block validation.

4. Experimental Setup

The CBDTF's effectiveness in mitigating DT within SG was evaluated using a real-world dataset, robust hardware setup, and a carefully selected software environment in a comprehensive experimental setup.

4.1 Dataset

The study used the Synthetic Models for Advanced, Realistic Testing: Distribution Systems and Scenarios (SMART-DS) dataset, developed by the National Renewable Energy Laboratory, to simulate real-world electrical distribution systems. The dataset, which includes data from San Francisco, Greensboro, and Austin, includes detailed network topologies and 15-minute interval time-series data. It also includes RE profiles representing solar and wind generation, as well as end-use load profiles segmented by building types and consumption types. This granularity enables comprehensive testing of the CBDTF in environments that resemble actual solar generation operations.

4.2 Hardware and Software Specifications

The experiments were conducted on a high-performance computing cluster. Each compute node was equipped with dual Intel Xeon E5-2690 v4 processors (2.6 GHz, 14 cores per processor), 128 GB of DDR4 RAM, and 1 TB of SSD storage. A dedicated Gigabit Ethernet switch was used to enable low-EED communication among the nodes, ensuring efficient operation of the private BT. Each node in the cluster functioned as an independent BT user, collectively forming a distributed ledger set-up representative of SG stakeholders such as utility providers, consumers, and prosumers.

The software stack was meticulously configured to ensure compatibility and robustness. Ubuntu 20.04 LTS was selected as the operating system for its stability and extensive support for BT development. Hyperledger Fabric v2.2 was the BT platform, enabling permissioned BT features for secure and traceable data management. Chaincode written in Go was deployed to execute SC for data validation, traceability, and consensus operations. Apache Kafka was used for real-time data ingestion and processing, integrating with SMART-DS high-velocity data streams. PostgreSQL was used for metadata storage, facilitating efficient querying and analysis. Docker containers encapsulated components for consistency. The experiment involved ingesting sensor data from the SMART-DS into the BT network, which was then distributed to BC nodes via Apache Kafka. Hyperledger Fabric SC validated the data against predefined criteria, ensuring authenticity and accuracy. The data was recorded on the BC, embedding a cryptographic hash, timestamp, and validator's signature, creating an immutable audit trail for end-to-end traceability and prompt detection of DT attacks.

Table 1: Dataset Description

Feature	Description	Unit	Resolution
Region	The geographical area represented in the dataset (*e.g.,* San Francisco, Greensboro, Austin).	-	-
Network Topology	Details of substations, feeders, transformers, and customer connections in the distribution network.	-	High
Real Power (P)	Active power consumption and generation in the distribution system.	kW	15-Minute intervals
Reactive Power (Q)	Reactive power flow in the distribution network.	kVAR	15-minute intervals
Voltage	Voltage measurements at various nodes in the network.	V	15-Minute intervals
Current	Current measurements across distribution lines and nodes.	A	15-Minute intervals
Load Profiles	Granular breakdown of energy consumption by different building types and end-use categories.	kWh	15-Minute intervals
Renewable Energy Profiles	Solar and wind energy generation data with temporal and spatial variations.	kW	High-resolution temporal
Weather Data	Meteorological data, including temperature, wind speed, and solar energy correlated with the grid	°C, m/s, W/m²	15-Minute intervals

4.3 Attack Simulation Using SMART-DS

The proposed CBDTF's robustness was tested using the SMART-DS, which provides high-resolution data from energy distribution networks. The dataset's granularity and diversity enabled the generation of adversarial scenarios to test the model's resilience against data tampering (DT), False Data Injection (FDI), Sybil attacks, and other malicious activities. The simulations also included data manipulation and BT integration.

2. **Unauthorized Data Modification Attack (UDMA):** The UDMA involved altering specific data entries after they were recorded on the BT. For example, voltage capacities $'V(t)'$ from the SMART-DS were tampered with by introducing deviations $'\Delta V'$, generating new values $V'(t) = V(t) + \Delta V$. The simulation tested the immutability of the BT and its ability to detect changes. DT caused mismatches in cryptographic hashes, invalidating blocks and propagating conflicts throughout the blockchain, ensuring that validators promptly flagged any modifications.
3. **False Data Injection Attack (FDIA):** FDIA introduced invented data points into SMART-DS, simulated extreme conditions, and injected them into the BC before

ingestion, resulting in unrealistic spikes in Energy Consumption (EC) or RE generation. *e.g.,* solar power generation $P_{\mathrm{Solar}}\ (t) > 0$ was inserted for nighttime intervals, violating natural constraints. The BC-SC validation mechanisms successfully identified anomalies by cross-checking against temporal and physical constraints. Range checks, such as $P_{\mathrm{Solar}}\ (t) \in [0, P_{\mathrm{Max}}]$, and correlations with meteorological data prevented these falsified entries from being recorded on the BC.

3 **Sybil Attack:** A Sybil attack was simulated by presenting multiple adversarial nodes to the BC, who attempted to approve a DT block containing false SMART-DS. The DPoS consensus mechanism mitigated the attack by limiting the impact of malicious nodes. Validators were selected based on reputation and voting weight, highlighting the importance of the higher threshold $'\theta'$ in maintaining consensus integrity. Despite the presence of Sybil nodes, the system maintained fault tolerance and continued to operate securely.

4 **Denial-of-Service (DoS):** The system's resilience was tested by simulating a DoS attack by injecting a large volume of redundant transactions from the SMART-DS. The queuing system, implemented using Apache Kafka, prioritized valid transactions and efficiently managed the improved load. MT and EED metrics were monitored to prove the system's operational stability even under attack.

5 **Data Replay Attack:** Replay attacks were simulated by resending valid transactions from the SMART-DS to manipulate network outputs, such as energy billing or load prediction. The BT's SC logic detected duplicates by validating transaction hashes and timestamps, ensuring no transaction could be reused, Eq. (33)

$$H(T_i) \neq H(T_j), \tag{33}$$

Where,

- $T_i, T_j \rightarrow$ separate transactions. The immutability of the BT further prevented unauthorized additions of duplicate entries.

The SMART-DS was ingested into the BC in real-time, with each data point processed through the following pipeline:

a) Data ingestion using Apache Kafka to simulate high-velocity streams.
b) Validation of dataset-derived transactions using SC implemented on Hyperledger Fabric.
c) Cryptographic hashing and block formation for validated transactions.
d) Consensus-driven validation and recording of blocks in the distributed ledger.

4.4 Evaluation Metrics and Baseline Models

The performance of the proposed CBDTF was thoroughly evaluated using a set of quantitative metrics and compared against baseline models commonly employed for data integrity and security in distributed systems. These metrics were selected to measure the model's effectiveness in ensuring data integrity, resilience against attacks, and computational efficiency.

i) Evaluation Metrics

- **Detection Rate (DR):** The DR measures the model's ability to detect DT or falsified data. It is computed as the ratio of successfully detected attacks to the sum of attempted attacks, Eq. (34).

$$\mathrm{DR} = \frac{\text{Number of Detected Attacks}}{\text{Total Number of Attacks}}. \quad (34)$$

A higher DR indicates better system reliability.

- **False Positive Rate (FPR):** This metric quantifies the proportion of legitimate transactions incorrectly flagged as tampered, as shown in Eq. (35).

$$\mathrm{FPR} = \frac{\text{Number of Incorrectly Flagged Transactions}}{\text{Total Number of Legitimate Transactions}} \quad (35)$$

  - A low FPR is critical to minimize disruptions to normal operations.

- **Consensus Resilience (CR):** Consensus resilience evaluates the robustness of the DPoS mechanism under adversarial conditions, particularly against Sybil attacks. It measures the minimum percentage of malicious validators required to disrupt consensus, as shown in Eq. (36).

$$\mathrm{CR} = \frac{\text{Number of Compromised Validators}}{\text{Total Validators}} \times 100 \quad (36)$$

  - A higher value indicates stronger fault tolerance.

- **EED (L):** Validation and recording a block under normal and adversarial conditions is a key performance metric. EED is measured in milliseconds (*ms*), Eq. (37).

$$L = \text{Time Taken to Validate a Block}. \quad (37)$$

  - Maintaining low EED is critical for real-time SG applications.

- **NT (NT):** NT measures the number of *tx/kWh* by the BT, as shown in Eq. (38).

$$\mathrm{TP} = \frac{\text{Total Transactions Processed}}{\text{Total Time Taken (s)}} \quad (38)$$

  - A higher NT ensures scalability for handling large data sets, which is typical in SG environments.

- **EE:** The EE metric quantifies the EC during block validation and consensus processes:

$$\text{EE} = \frac{\text{Transactions Processed}}{\text{Energy Consumed (kWh)}} \tag{39}$$

Higher EE is significant in RE systems, such as SG.

- **Tamper Resistance Index (TRI):** This index measures the model's ability to resist DT attempts, integrating the DR and FPR:

$$\text{TRI} = \frac{\text{DR}}{\text{FPR}+\epsilon},\ \epsilon > 0 \tag{40}$$

  - A higher TRI value indicates superior DT resistance.

ii) **Baseline Models:** The proposed CBDTF was compared against several established baseline models to prove its security, efficiency, and scalability advantages.

- **PoW-BC:** PoW, like Bitcoin, was used as a baseline for DT resistance and security. While PoW provides strong immutability guarantees, it suffers from high EC and low NT, making it unsuitable for real-time SG applications.
- **DPoS-BC:** DPoS was evaluated for EE compared to PoW. However, DPoS mechanisms frequently challenge scalability and decentralization, particularly in adversarial scenarios such as Sybil attacks.
- **PBFT:** PBFT, commonly used in permissioned BC, served as a baseline for low-EED and high-NT consensus. Its performance degrades in larger networks, highlighting its limitations in highly distributed SG environments.
- **Centralized Database Systems (CDS) (No BC):** Traditional centralized database models were included for comparison of data traceability and DT resistance. While these systems propose high NT, they lack the immutability and transparency that BT provides, making them vulnerable to insider threats and data theft.
- **Hybrid PoW-PoS-BC:** Hybrid models combining PoW and PoS mechanisms were used to benchmark the EE and security trade-offs. These systems verified moderate performance but were outperformed by DPoS in terms of scalability and EED.

4.5 BC Network Configuration

The BC for CBDTF implementation adopts a permissioned architecture based on Hyperledger Fabric, incorporating multiple organizations representing different SG stakeholders. The network topology establishes a distributed network where each organization maintains peer nodes that participate in transaction validation and block formation. This configuration implements Byzantine fault tolerance NT carefully defined endorsement policies requiring signatures from a minimum of '*k*' out of '*n*' organizations, where $k = \lceil 2n/3 \rceil$.

The network implements a multi-channel configuration to segregate different grid measurements, with each channel maintaining its ledger. Critical data streams such as power measurements and voltage readings require higher endorsement thresholds (75% of organizations) than routine configuration updates (51% of organizations). Private data collections enable selective data sharing among organizations while maintaining confidentiality through cryptographic hashing of shared data between organization pairs.

Table 2: Blockchain Network Configuration Parameters

Parameters	Configuration Detail	Value
Block Parameters	Maximum Block Size	2 MB
	Block Generation Time	5 Seconds
	Maximum Transaction Size	512 KB
State Database	Cache Size per Peer	[illegible] MB
	Database Size per Channel	1 GB
	Database Type	CouchDB
Ordering Service	Number of Nodes	5
	Consensus Protocol	Raft
	Batch Timeout	2 Seconds
	Maximum Message Count	500
Certificate Authority	Key Size	2048 Bits
	Validity Period	365 Days
	Signature Algorithm	ECDSA-SHA256
	Max Enrollments/Identity	5
Gossip Protocol	Alive Time Interval	5 Seconds
	Expiration Timeout	25 Seconds
	Reconnect Interval	25 Seconds
	Max Block Distance	20

The ordering service utilizes a Raft-based consensus mechanism with five ordering nodes distributed across organizations. The block-cutting parameters are optimized for optimal performance and network stability, with each organization having its own Certificate Authority (CA) that adheres to standardized security parameters. The gossip protocol parameters ensure efficient peer-to-peer communication and block propagation. The state database configuration utilizes CouchDB with optimized cache and storage parameters to facilitate efficient query operations and promote reasonable resource utilization. Network parameters are continuously monitored by the configuration service, allowing for dynamic adjustments based on performance metrics and operational requirements.

5. Results and Analysis

5.1 Detection Rate

The proposed CBDTF demonstrated exceptional performance in mitigating numerous attack scenarios, with an average DR of 98.7%. Its robust cryptographic validation mechanisms and advanced traceability features outperformed baseline models. Its effectiveness was particularly notable in Unauthorized Data Modification and Data Replay, where its hash-based integrity checks and SC rules ensured near-complete DR of DT transactions (Figure 4).

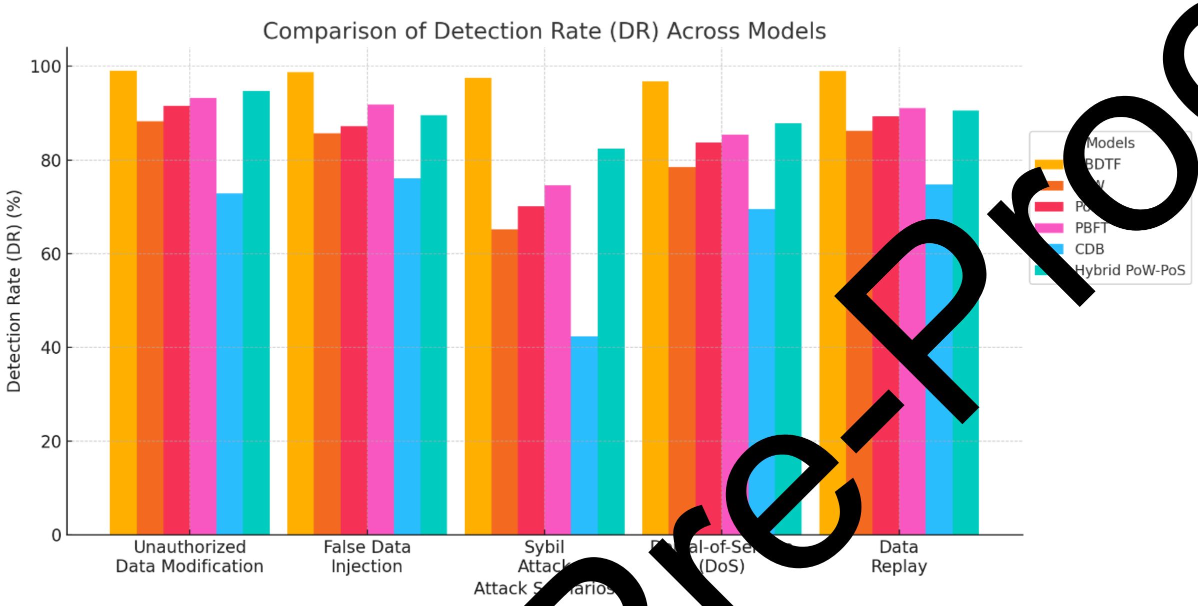


Figure 4: DR analysis

PoW and DPoS demonstrated moderate DR capabilities, with average DR rates of 80.8% and 84.4%, respectively. PoW's computationally intensive validation process effectively combats DT attacks, but Sybil attacks pose challenges due to its lack of identity verification mechanisms. PoS, on the other hand, outperforms PoW in most scenarios but has vulnerabilities in Sybil

The Practical Byzantine Fault Tolerance (PBFT) achieved a competitive average DR of 87.2%, using a deterministic consensus mechanism for strong DT as DR. However, its scalability challenges reduced performance under high-load attacks, such as DoS. The CDS performed poorly, with an average DR of 67.1%, due to its lack of distributed validation and single point of control.

The Hybrid PoW-PoS system effectively balances PoW + PoS, achieving an average DR of 89.0%. It was effective against Data Replay but fell short of CBDTF in scenarios requiring higher precision and DT resistance.

The Hybrid PoW-PoS system effectively balanced PoW + PoS, achieving an average DR of 89.0%. It was effective against Data Replay but fell short of CBDTF in scenarios requiring higher precision and DT resistance.

5.2 False Positive Rate (FPR)

The analysis of FPR (Figure 5) reveals that models with lower FPR can effectively detect DT transactions and distinguish legitimate and malicious data without disrupting normal operations.

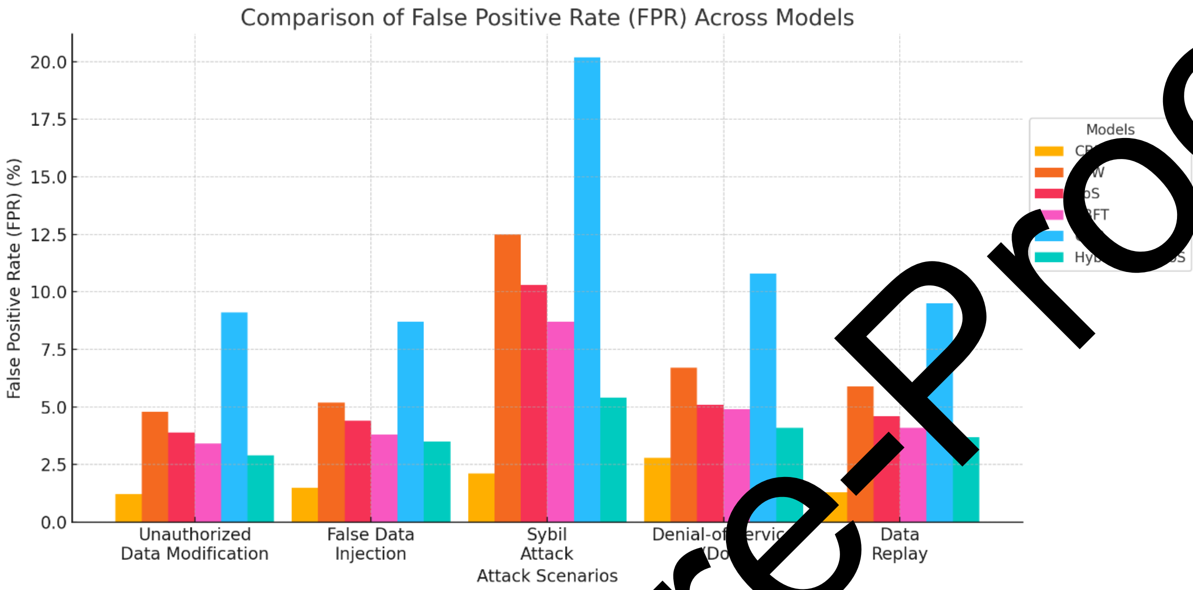


Figure 5: FPR analysis

CBDTF has the lowest FPR across all attack scenarios, averaging 1.78%, primarily due to its robust validation mechanisms. It performs well in Unauthorized Data Modification (UDM) and Data Replay, with an FPR below 3% in challenging scenarios like Sybil and DoS. PoW, on the other hand, has a higher FPR of 7.02% due to its computational mining process, which lacks nuanced validation mechanisms. It challenges in detecting Sybil, with an FPR of 12.5%. While its FPR is lower for simpler scenarios, it is less reliable than CBDTF.

PoS improved over PoW with an average FPR of 5.66% but displayed vulnerabilities in Sybil. PoS demonstrated stable FPR values in scenarios such as False Data Injection (FDI) and Data Replay, thanks to stake-based validation. PBFT exhibited a balanced performance, with an average FPR of 5.00%, and deterministic consensus effectively reduced false alarms in scenarios such as UDM and FDI. However, scalability issues under Sybil and DoS resulted in slightly higher FPR.

The centralized model had the highest average FPR of 11.66%, but its vulnerability in adversarial conditions was evident. It was prone to frequent misclassification, particularly in Sybil and DoS, indicating its inability to maintain reliable validation under attack. The hybrid model achieved an average FPR of 3.92%. It proved consistent performance across all scenarios, with the lowest FPR recorded in UDM (FPR: 2.9%) and the highest in Sybil (FPR: 5.4%).

5.3 Consensus Resilience (CR) Across

Figure 6 illustrates the CR of the proposed CBDTF and baseline models across numerous attack scenarios. CR measures the robustness of a consensus mechanism under adversarial conditions, reflecting the ability to maintain data integrity and operational stability.

With an average CR of 96.2%, the CBDTF outperformed all baseline models across all attack scenarios. The DPoS consensus mechanism proved highly effective in resisting adversarial attacks, particularly in scenarios like Data Replay (CR: 98.1%) and UDM (CR: 97.5%). The model verified slight reductions in resilience under Sybil (CR: 95.2%) and DoS (CR: 93.4%), but its performance remained consistently high, showcasing its robustness. PoW achieved an average CR of 84.3%, demonstrating moderate resilience in UDM (91.3%) and FDI (89.5%). However, its resilience significantly dropped in Sybil (CR: 67.8%) due to the absence of identity validation mechanisms—the model challenge under DoS scenarios, where high computational overhead impeded performance.

PoS performed slightly better than PoW, with an average CR of 86.6%. It maintained strong resilience in scenarios like Data Replay (CR: 90.1%) and FDI (CR: 91.4%). However, it showed reduced resilience under Sybil (CR: 72.5%), highlighting vulnerabilities in its validator selection process when adversaries compromised stakes. PBFT verified consistent performance with an average CR of 89.9%, excelling in UDM (CR: 94.2%) and Data Replay (CR: 94.3%). The deterministic nature of PBFT provided robust DT resistance, but its limited scalability reduced its resilience in larger networks, particularly during Sybil (CR: 80.1%) and DoS (CR: 87.4%).

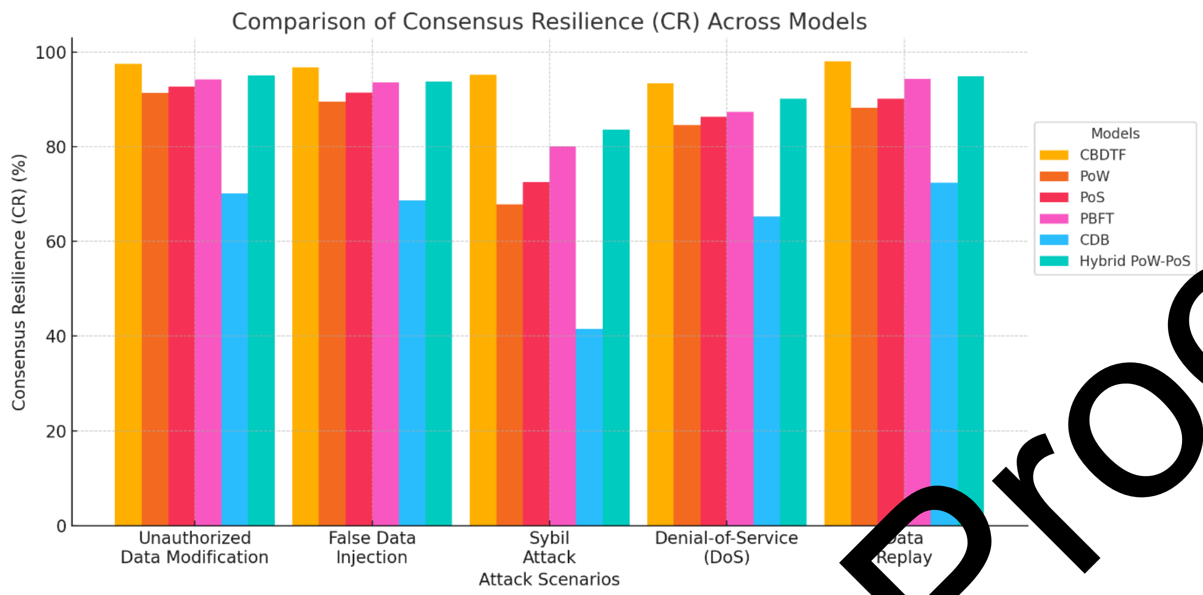


Figure 6: Consensus Resilience analysis

The centralized model recorded the lowest average CR of 63.6%, highlighting its vulnerabilities in all scenarios. It performed poorly in Sybil (CR: 41.5%) and DoS (CR: 65.3%) due to the lack of distributed validation and redundancy. While marginally better in simpler scenarios, such as UDM (CR: 70.2%), it remains unsuitable for adversarial environments. The hybrid model achieved an average CR of 91.5%, combining the strengths of PoW and PoS. It performed well across all scenarios, particularly in Data Replay (94.8%) and UDM (CR: 95.1%). However, its resilience under Sybil (CR: 83.6%) was lower than CBDTF.

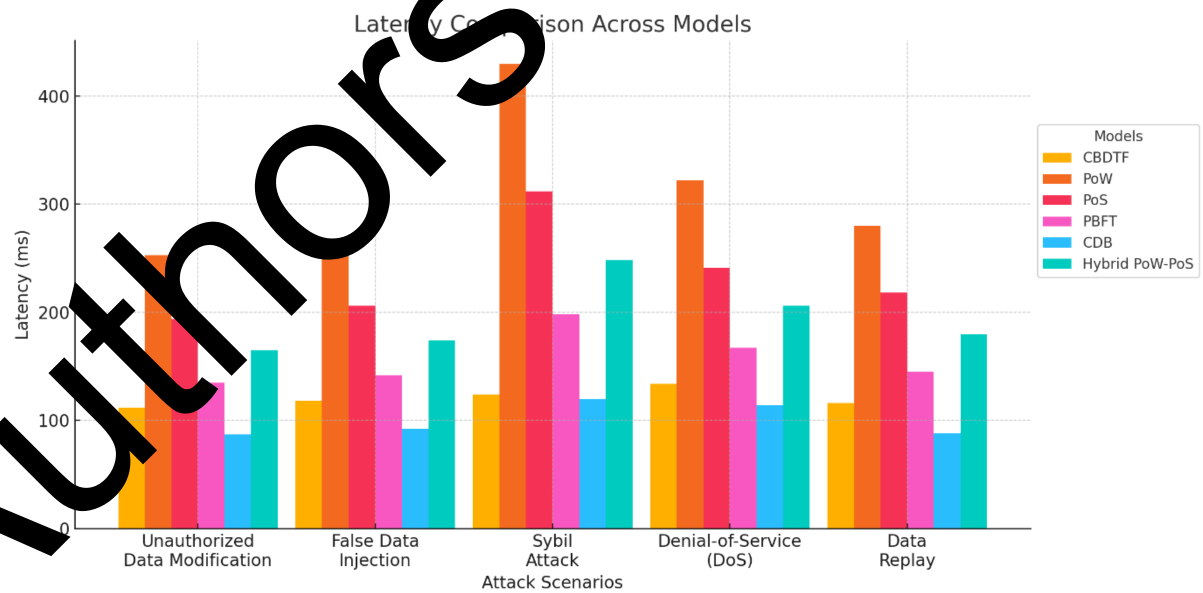


Figure 7: EED analysis

5.4 EED (*ms*) Comparison

EED measures the time it takes for a model to validate and process a block, highlighting its efficiency in real-time operations. Figure 7 demonstrates the EED performance of the proposed CBDTF and baseline models. The CBDTF achieved an average EED of 120.8 *ms*, showcasing its efficiency in handling real-time transactions. Its low EED across all attack scenarios, particularly in UDM (112 *ms*) and Data Replay (116 *ms*), is attributed to the lightweight DPoS mechanism. PoW recorded the highest average EED at 309.8 *ms*, with severe delays under Sybil (430 *ms*). The computationally intensive mining process significantly increased validation times, making it unsuitable for real-time applications. PoS improved upon PoW, achieving an average EED of 234.2 *ms*. However, its performance declined under Sybil (312 *ms*) and DoS scenarios (241 *ms*) due to the overheads associated with stake-based validation. PBFT proved moderate EED (average 157.4 *ms*), performing well in Unauthorized Data Modification (135 *ms*) and Data Replay Attacks (145 *ms*). However, the deterministic consensus mechanism added delays in more extensive networks under Sybil (198 *ms*). Due to its non-distributed architecture, the centralized model achieved the lowest EED (average 100.2 *ms*). However, the absence of decentralization compromises its security, making it inappropriate for adversarial conditions. The hybrid model balanced the cryptographic robustness of PoW and the efficiency of PoS, achieving an average EED of 194.6 *ms*. Its EED under Sybil (248 *ms*) and DoS scenarios (2[illegible] *ms*) was higher than that of CBDTF but lower than that of PoW.

Figure 8: NT analysis

5.5 NT Comparison

NT measures the number of Transactions Processed Per Second (TPS), reflecting the scalability of each model. Figure 8 highlights the NT performance across all models. The CBDTF achieved an average NT of 1113 TPS, making it the most efficient decentralized model. It performed consistently well across all scenarios, with exceptionally high NT in UDM (1213 TPS) and Data Replay (1182 TPS). PoW recorded the lowest average NT at 279.2 TPS, with significant drops under Sybil Attacks (208 TPS). Its reliance on mining reduced TPS, limiting its scalability. PoS improved NT compared to PoW, achieving an average of 594 TPS. It maintained stable performance in most scenarios but exhibited reduced NT under Sybil (4[illegible]9 TPS) due to validator inefficiencies. PBFT achieved an average NT of 861 TPS, benefiting from its efficient consensus mechanism in smaller networks. Its performance declined under high-load scenarios, such as DoS (813 TPS). The centralized model achieved the highest average NT at 1541.6 TPS, signifying its advantage in non-distributed environments. However, it lacks the security and fault tolerance necessary for DT-resistant systems. The hybrid model achieved an average NT of 800.4 TPS, balancing the strengths of PoW and PoS. Its performance was consistent across scenarios but lagged behind CBDTF due to its higher validation complexity.

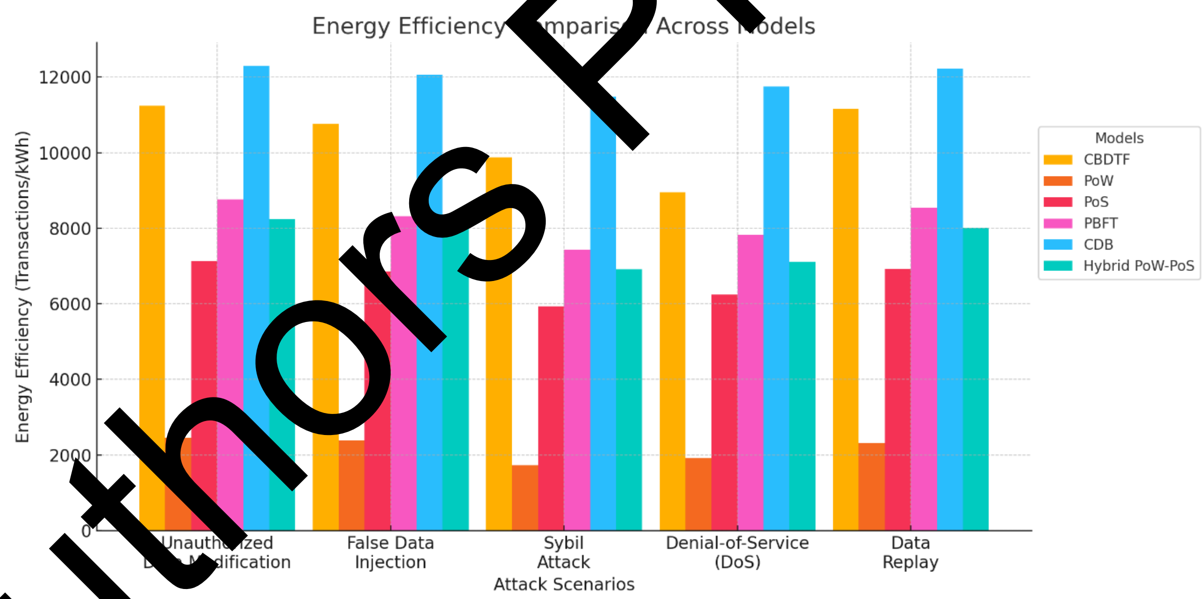


Figure 9: EE analysis

7.6 EE (transactions per kilowatt-hour (tx/kWh))

Figure 9 compares EE and highlights the operational sustainability of the proposed CBDTF and baseline models.

The CBDTF achieved an average EE of 10,395.6 *tx/kWh*, ranking second among all models. Its lightweight DPoS mechanism minimizes computational overhead while maintaining high NT, resulting in superior performance under scenarios such as UDM (11,237 *tx/kWh*) and Data Replay (11,162 *tx/kWh*). Due to its non-distributed architecture, the CDS achieved the highest EE of 11,964.6 *tx/kWh*; however, this efficiency comes at the cost of reduced DT resistance and resilience to adversarial attacks.

PBFT verified strong EE with an average of 8,180.2 *tx/kWh*, leveraging its deterministic consensus mechanism. However, its performance declined slightly in adversarial scenarios, such as Sybil (7,437 *tx/kWh*). The hybrid model balanced the strengths of PoW and PoS, achieving an average of 7,667.2 *tx/kWh*. Its performance was consistent across all scenarios, with the highest efficiency in Data Replay (8,016 *tx/kWh*). PoS averaged 6,621.[illegible] *tx/kWh*, with lower EE under high-load scenarios like DoS (6,247 *tx/kWh*). Its efficiency was better than PoW but inferior to CBDTF. PoW recorded the lowest EE at 2,1[illegible]8.6 *tx/kWh*, reflecting the high computational cost of mining. Its performance under scenarios like Sybil (1,728 *tx/kWh*) further highlighted its unsuitability for energy-sensitive applications.

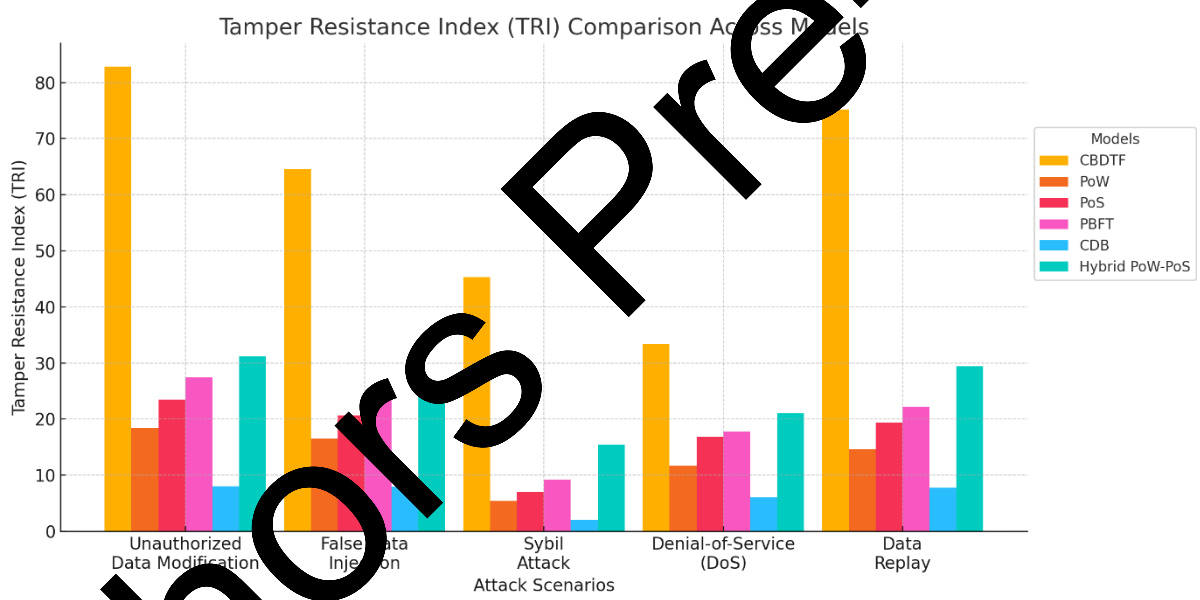


Figure 10: TRI analysis

5.7 Tamper Resistance Index (TRI)

The TRI (Figure 10) assesses the models' ability to resist DT while minimizing FP and maintaining high DR accuracy. The CBDTF achieved the highest average TRI of 60.26, significantly outperforming all baseline models. Its superior performance across scenarios, such as UDM (82.9) and Data Replay (75.2), underscores its robust validation mechanisms and cryptographic security. The hybrid model achieved the second-highest average TRI at 24.7, performing well in scenarios like UDM (31.2). Its combination of PoW's immutability and PoS's efficiency provided balanced DT resistance. PBFT recorded an average TRI of 20.24, benefiting from its deterministic consensus. However, its limited scalability reduced its effectiveness in adversarial scenarios, such as Sybil (9.2). PoS achieved an average TRI of 17.5, performing consistently better than PoW in most scenarios. Its performance in Data Replay (19.4) highlights its stake-based validation strengths. PoW exhibited an average TRI of 13.32, reflecting its vulnerabilities in scenarios like Sybil (5.4). Its high computational demands further constrained its DT resistance. The CDS had the lowest TRI at 6.36, demonstrating significant weaknesses in adversarial conditions. Its inability to handle distributed validation made it highly susceptible to DT.

6. Conclusion and Future Work

The CBDTF is a comprehensive model that effectively mitigates DT attacks in SG environments. Its multi-layered network integrates BT+ SG operations, providing robust security without compromising performance. The model's DR of 98.7% across various attack scenarios and low FPR of 1.78% prove its superior ability to identify and prevent DT attempts, advancing *state-of-the-art* SG security. The DPoS consensus mechanism has demonstrated 96.2% resilience and an EED of 120.8 *ms*, outperforming traditional BT, making real-time data validation feasible in SG operations. The model can process 1,113 *tx/kWh* while maintaining an EE of 10,395.6 *tx/kWh*, making it practical for large-scale deployment. CBDTF's success validates the effectiveness of community-driven validation in enhancing security and reducing computational overhead, setting new benchmarks for BT-based security solutions in critical setup security. The integration of DLT in SG operations proposes a blueprint for securing other critical systems. However, further investigation is required to validate its scalability and availability against emerging attack vectors and zero-day exploits as technology evolves, as well as to test it with more extensive networks.

Future research should focus on advanced Machine Learning for enhanced attack detection, dynamic network consensus mechanisms, cross-chain interoperability, privacy-preserving features, and quantum-resistant cryptographic protocols for improved grid coordination.

References

[1] Abdelkader, S., Amissah, J., Kinga, S., Mugerwa, G., Emmanuel, E., Mansour, D. E. A., ... & Prokop, L. (2024). Securing modern power systems: Implementing comprehensive strategies to enhance resilience and reliability against cyber-attacks. Results in engineering, 102647.

[2] Diaba, S. Y., Shafie-khah, M., & Elmusrati, M. (2024). Cyber-physical attack and the future energy systems: A review. Energy Reports, 12, 2914-2932.

[3] Rajkumar, V. S., Ştefanov, A., Presekal, A., Palensky, P., & Torres, J. L. R. (2023). Cyber-attacks on power grids: Causes and propagation of cascading failures. IEEE Access.

[4] Zhao, A. P., Li, S., Gu, C., Yan, X., Hu, P. J. H., Wang, Z., ... & Hernando-Gil, I. (2024). Cyber vulnerabilities of energy systems. IEEE Journal of Emerging and Selected Topics in Industrial Electronics.

[5] Ali, S. S., & Choi, B. J. (2020). State-of-the-art artificial intelligence techniques for distributed smart grids: A review. Electronics, 9(6), 1030.

[6] Kabalci, E., & Kabalci, Y. (Eds.). (2019). Smart grids and their communication systems (No. 1). Singapore: Springer.

[7] Liu, M., Teng, F., Zhang, Z., Ge, P., Sun, M., Deng, R., ... & Chen, J. (2024). Enhancing cyber-resiliency of DER-based smart grid: A survey. IEEE Transactions on Smart Grid.

[8] Ghiasi, M., Niknam, T., Wang, Z., Mehrandezh, M., Dehghani, M., & Ghadimi, N. (2023). A comprehensive review of cyber-attacks and defense mechanisms for improving security in smart grid energy systems: Past, present and future. Electric Power Systems Research, 215, 108975.

[9] Kumar, P., Lin, Y., Bai, G., Paverd, A., Dong, J. S., & Martin, A. (2019). Smart grid metering networks: A survey on security, privacy and open research issues. IEEE Communications Surveys & Tutorials, 21(3), 2886-2927.

[10] Baba, M., Nor, N., Sheikh, A., Nowakowski, G., Masood, F., Rehman, M., ... & Mohan, B. (2022). A review of the importance of synchrophasor technology, smart grid, and applications. Bulletin of the Polish Academy of Sciences Technical Sciences, e143826-e143826.

[11] Chen, Y., Kumara, E. K., & Sivakumar, V. (2021). Invesitigation on finance industry on risk awareness model and digital economic growth. Annals of Operations Research, 1-22.

[12] Abdelkader, S., Amissah, J., Kinga, S., Mugerwa, G., Emmanuel, E., Mansour, D. E. A., ... & Prokop, L. (2024). Securing modern power systems: Implementing comprehensive strategies to enhance resilience and reliability against cyber-attacks. Results in engineering, 102647.

[13] Nguyen, L. H., Nguyen, V. L., Hwang, R. H., Kuo, J. J., Chen, Y. W., Huang, C. C., & Pan, P. I. (2024). Towards Secured Smart Grid 2.0: Exploring Security Threats, Protection Models, and Challenges. IEEE Communications Surveys & Tutorials.

[14] Abd Algani, Y. M., Rao, V. S., & Saravanakumar, R. (2024, June). AI-Powered Secure Decentralized Energy Transactions in Smart Grids: Enhancing Security and Efficiency. In 2024 IEEE 3rd International Conference on Electrical Power and Energy Systems (ICEPES) (pp. 1-5). IEEE.

[15] Zhang, M., Liu, Y., Cheng, Q. et al. Smart grid security based on blockchain and smart contract. Peer-to-Peer Netw. Appl. **17**, 2167–2184 (2024).

[16] https://smartgrid.ieee.org/bulletins/july-2018/is-the-blockchain-a-good-solution-for-cybersecurity-in-the-smart-grid

[17] S. Alghasabi, A. Shaf, T. Ali, M. Zafar, M. Irfan, and T. Alsuwian, "Securing Smart Grid Data With Blockchain and Wireless Sensor Networks: A Collaborative Approach," in IEEE Access, vol. 12, pp. 19181-19198, 2024,

[18] Bolgouras, V., Ioannidis, T., Politis, I., Zarras, A., & Xenakis, C. (2024). RETINA: Distributed and secure trust management for smart grid applications and energy trading. Sustainable Energy, Grids and Networks, 38, 101274.

[19] Reijsbergen, D., Maw, A., Dinh, T. T. A., Li, W. T., & Yuen, C. (2022, April). Securing smart grids through an incentive mechanism for blockchain-based data sharing. In Proceedings of the Twelfth ACM Conference on Data and Application Security and Privacy (pp. 191-202).

[20] Guo, Y., Wan, Z., & Cheng, X. (2022). When blockchain meets smart grids: A comprehensive survey. High-Confidence Computing, 2(2), 100059.