ChainGrid-Offline: A TEE-Backed Protocol for Cascaded Offline Smart Grid Payments

Ali Aouto, Jae-Min Lee and Dong-Seong Kim
Networked Systems Lab., Dept. of IT Convergence Engineering,
Kumoh National Institute of Technology, Gumi, South Korea.
Email: {ali.aouto,ljmpaul,dskim}@kumoh.ac.kr

***Abstract*—ChainGrid-Offline enables secure, cascaded offline payments for decentralized smart grids using Trusted Execution Environments (TEEs). It introduces ephemeral key attestations, Merkle-DAG transaction logs, and federated learning for demand-aware token issuance. Simulated across 1,200 transactions and 50 nodes, the system demonstrates low latency, strong security, and seamless reconciliation under disconnected conditions.**

***Index Terms*—TEE-secured Offline Payments, Merkle-DAG Reconciliation, Decentralized Energy Trading**

I. Introduction

The decentralization of energy systems has transformed consumers into "prosumers" through rooftop solar panels, EVs, and local microgrids. This trend calls for peer-to-peer (P2P) energy trading platforms that function independently of centralized infrastructure [1].

While blockchain enables secure and transparent trading, it generally assumes reliable connectivity and high computational capacity—both of which are absent in rural or disaster-prone regions [2]. Protocols like Lightning Network and Raiden rely on pre-funded channels and bi-directional locks, making them unsuitable for ad-hoc or multi-hop transactions in disconnected settings [3].

ChainGrid-Offline introduces a TEE-based protocol for verifiable, offline, and cascaded energy payments. Each node operates within a Trusted Execution Environment (TEE), managing private keys and sealing transaction logs. Transactions use ephemeral keys signed by a persistent TEE root identity and are organized into a Merkle-DAG for future on-chain reconciliation.

Our system enables offline multi-hop payments—users can reuse received tokens for downstream transfers, mimicking the flexibility of cash. This feature is critical in environments such as solar microgrids and emergency charging setups.

We simulate over 1,000 transactions across 50 nodes, evaluating latency, reconciliation accuracy, and security. Results show cryptographic efficiency, sublinear latency growth, and resilience against attacks like forgery and double-spending.

Contributions:

- Design of ChainGrid-Offline: a TEE-secured, offline, multi-hop payment protocol.
- DAG-based logging with Merkle root reconciliation for auditability and chain updates.
- Evaluation over realistic energy trading scenarios with high security guarantees.

II. Proposed Model

ChainGrid-Offline enables secure, offline, and cascaded energy payments in disconnected environments using TEE-protected devices and Merkle-DAG logs.

A. System Architecture

The system consists of:

- **TEE Node:** Manages keys, signs transactions, and seals logs.
- **Ephemeral Keys:** One-time key pairs (pk^t, sk^t), signed by a persistent TEE root key.
- **Merkle-DAG:** Transactions are hashed and linked, supporting flexible cascades.
- **Smart Contract:** Accepts Merkle roots and inclusion proofs for on-chain updates.

B. Transaction Format

Each transaction is:

$$tx_i = (pk_i^t, addr_j, \delta E, T_i, M_i, \sigma_i, \sigma_{auth})$$

where σ_i is signed by the ephemeral key, and σ_{auth} authenticates it via the root key.

C. Offline Transaction Protocol

ChainGrid-Offline enables secure, cascaded payments using TEEs and ephemeral keys. The protocol consists of three main stages:

1) Initialization and Deposit: Each user initializes a root key pair (pk^{root}, sk^{root}) inside their TEE, sets the balance to zero, and starts an empty Merkle-DAG log. When a user deposits tokens via the smart contract, the deposit is logged and the local balance is updated within the enclave.

2) Offline Transfer: To transfer tokens offline, the sender generates a fresh ephemeral key pair (pk^t, sk^t), signs it with their root key to create an attestation σ_{auth}, and uses the ephemeral private key to sign the transaction. The signed transaction and attestation are appended to the sender's DAG and transmitted to the recipient. The recipient verifies both signatures inside their TEE, ensures the transaction is unique and timely, then updates their balance and adds the transaction to their own DAG.

3) Cascaded Reuse: After receiving a valid payment, a user can immediately reuse those tokens by issuing a new

transaction. A new ephemeral key is generated and signed, forming a link to the previous transaction in the DAG. This chaining allows for multi-hop token forwarding without any need for online interaction, maintaining full traceability and auditability.

All transactions are eventually reconciled on-chain via Merkle root submission, proving the integrity of the offline DAGs without uploading their full content.

D. Unified Offline Transaction Protocol

Algorithm 1 ChainGrid-Offline Unified Protocol

procedure INITTEE
  Generate root keypair (pk^{root}, sk^{root})
  Initialize balance $B \leftarrow 0$, DAG log $G \leftarrow \emptyset$
  Store manufacturer key for attestation
end procedure
procedure DEPOSIT(δE)
  Submit deposit to smart contract
  On confirmation, update $B \leftarrow B + \delta E$, log deposit in G
end procedure
procedure OFFLINETRANSFER($addr_{recv}, \delta E, M$)
  Generate ephemeral keys (pk^t, sk^t)
  $\sigma_{auth} \leftarrow \text{Sign}_{sk^{root}}(pk^t)$
  Create transaction $tx \leftarrow (pk^t, addr_{recv}, \delta E, T, M)$
  $\sigma \leftarrow \text{Sign}_{sk^t}(tx)$
  Append $tx, \sigma, \sigma_{auth}$ to DAG G, update $B \leftarrow B - \delta E$
  Transmit $(tx, \sigma, \sigma_{auth})$ to recipient
end procedure
procedure RECEIVE($tx, \sigma, \sigma_{auth}$)
  Verify σ_{auth} with manufacturer key
  Verify σ using pk^t from tx
  if valid and $tx \notin G$ **then**
    Append to G, update $B \leftarrow B + \delta E$
  else
    Reject transaction
  end if
end procedure
procedure CASCADETRANSFER($addr_k, \delta E', tx_{prev}$)
  if $\delta E' > B$ **then**
    Abort
  end if
  Generate new keys $(pk^t_{\text{new}}, sk^t_{\text{new}})$
  $\sigma^{\text{new}}_{\text{auth}} \leftarrow \text{Sign}_{sk^{root}}(pk^t_{\text{new}})$
  Create new tx, sign with sk^t_{new}
  Link $tx_{prev} \rightarrow tx_{\text{new}}$ in DAG, update $B \leftarrow B - \delta E'$
  Transmit to $addr_k$
end procedure

III. Simulation Results and Discussion

We evaluated ChainGrid-Offline using 50 simulated TEE-enabled nodes across 1,200 offline transactions. Nodes operated under partial connectivity with randomized token values and cascade depths.

A. Simulation Setup

- **Nodes:** 50 TEE-capable prosumers
- **Disconnection Rate:** Up to 80% of session
- **Token Amount:** 1–5 units per transfer
- **Cascade Depth:** Randomized, 2–8 hops
- **Reconciliation:** Merkle root submission every 20 transactions

B. Performance Overview

The protocol demonstrated efficient cryptographic operations and accurate reconciliation. Table I highlights the key performance metrics.

TABLE I
CHAINGRID-OFFLINE PERFORMANCE METRICS

Metric	Average Value
Ephemeral Key Generation	2.7 ms
Transaction Signing	4.3 ms
Signature Verification	5.1 ms
DAG Storage per Transaction	1.2 KB
Cascade Latency (5 hops)	12.8 ms
Merkle Reconciliation (50 txs)	96.3 ms
Reconciliation Accuracy	100%

C. Security Validation

ChainGrid-Offline successfully mitigated common threats:

- **Replay attacks:** Blocked via timestamp checks.
- **Key forgery:** Prevented by TEE-signed attestations.
- **Log tampering:** Detected by Merkle root mismatch.

All transactions were verifiable, and audit integrity was fully preserved.

IV. Conclusion

ChainGrid-Offline enables secure, offline energy payments using TEEs, ephemeral keys, and Merkle-DAGs. It supports multi-hop token reuse and auditability without requiring constant connectivity. Simulations confirm low latency, linear scalability, and strong resistance to attacks, making it well-suited for decentralized energy trading in constrained environments.

Acknowledgment

This work was partly supported by Innovative Human Resource Development for Local Intellectualization program through the IITP grant funded by the Korea government(MSIT) (IITP-2025-RS-2020-II201612, 33%) and by Priority Research Centers Program through the NRF funded by the MEST(2018R1A6A1A03024003, 33%) and by the MSIT, Korea, under the ITRC support program(IITP-2025-RS-2024-00438430, 34%)

References

[1] A. Kumari, U. Chintukumar Sukharamwala, S. Tanwar, M. S. Raboaca, F. Alqahtani, A. Tolba, R. Sharma, I. Aschilean, and T. C. Mihaltan, "Blockchain-based peer-to-peer transactive energy management scheme for smart grid system," *Sensors*, vol. 22, no. 13, 2022.
[2] D.-S. Kim and R. Syamsul, "Integrating machine learning with proof-of-authority-and-association for dynamic signer selection in blockchain networks," *ICT Express*, vol. 11, no. 2, pp. 258–263, 2025.
[3] J. Poon and T. Dryja, "The bitcoin lightning network: Scalable off-chain instant payments," 2016.