**GhostCore Red Team Research Brief**

**Title**: Phantom UTXO Aggregation Exploit - Exploiting Blockchain Trust via Ghost Wallet Dust Harvesting

**Classification**: CERT Submission - Ethical Red Team Simulation **Authors**: GhostCore Systems Research Division **Date**: [Auto-Generated]

### **Executive Summary:**

This document details a simulated attack chain targeting blockchain infrastructure using phantom UTXOs (unspent transaction outputs) aggregated from dust transactions across the public ledger. This exploit is designed as an **ethical proof-of-concept** to demonstrate that **trust assumptions in the UTXO model can be manipulated off-chain** to deceive third-party systems such as Bitcoin ATMs, decentralized apps, and lightweight wallets.

### **Attack Name:**

**Phantom UTXO Aggregation Exploit**

### **Vector Class:**

* Physical-Digital Bridge
* Off-Chain Trust Manipulation
* Blockchain Metadata Abuse

### **Attack Chain Overview:**

1. **Dust UTXO Enumeration**:  
   * Scan public blockchain for UTXOs under a set threshold (e.g., < 600 sats).
   * Harvest metadata: txid, output\_n, address, value.
2. **Ghost Wallet Emulation**:  
   * Construct fake wallet balances using non-owned but existing UTXO metadata.
   * Never broadcast: transactions remain local.
3. **Synthetic Transaction Generation**:  
   * Aggregate dust UTXOs into a single high-value phantom transaction.
   * Signed with pre-generated key pair that never hits chain.
4. **Fake Verification API / UI Injection**:  
   * Display fabricated balances or "pending" confirmations.
   * Exploit ATM/mixer logic that only checks for value, not ownership.
5. **Exploit Trigger**:  
   * Initiate transaction or cashout via API/webhook.
   * If integrated with fiat systems, could result in withdrawal of real funds.

### **Red Team Ethical Use Cases:**

* **Penetration Test of Blockchain ATM Logic**:  
  + Determine if off-chain validation is robust.
  + Highlight weaknesses in 3rd party API trust.
* **Smart Contract Resilience Testing**:  
  + Simulate phantom inputs and observe contract logic.
* **Wallet UX Audit**:  
  + Detect if wallet balance rendering is exploitable.

### **Countermeasures:**

* Require **signature validation on balance fetch**, not just UTXO existence.
* Use **multi-source consensus** before accepting transaction as verified.
* Restrict ATM/API responses to **broadcast-only** transactions.
* Perform **chain of custody checks** on transaction history.

### **Ethical Notes:**

This exploit was developed **solely for defensive research and CERT outreach**. It has not been used for malicious purposes, and all test simulations occurred in isolated, non-public environments.

### **Visual Payload Simulation:**

# Simulated Phantom Transaction Creation

from bitcoinlib.transactions import Transaction

phantom\_tx = Transaction()

phantom\_tx.add\_input(prev\_txid='deadbeef', output\_n=0, value=420, address='1GhostPhantom...')

phantom\_tx.add\_output(9999999, '1RedTeamWallet...')

### **Appendix:**

* UTXO Dust Analysis Tool (pseudo-code)
* Mempool Ghost Transaction Visualizer
* Phantom Wallet Aggregator Module (internal only)

**Contact GhostCore CERT Division** at redteam@ghostcore.systems for further coordination or private briefings.

**END OF DOCUMENT**