



RESEARCH REPORT

Mobile Money Fraud: Control Patterns That Hold

How fraud actually enters mobile money and agent-banking platforms — social engineering, SIM swap, agent collusion, API abuse — and the control patterns that stand up to each.

PUBLISHED

October 7, 2025

READING TIME

13 minutes

AUTHOR

SCIAT AFRICA Research

ABSTRACT

Mobile money platforms are attacked at their human edges far more often than at their cryptography. This report walks the common fraud paths – customer social engineering, SIM swap, agent collusion, insider access and third-party API abuse – and sets out the detection and control patterns that have proved durable, written for product, risk and security teams building on these rails.

Topics: FinTech · Fraud · Application Security

WHAT TO TAKE AWAY

- The dominant loss paths are social and procedural, not cryptographic – controls aimed only at the protocol miss them.
- SIM swap defeats SMS as an authentication factor; binding sessions to a device changes the economics.
- Agent networks need transaction-pattern monitoring, because agent fraud looks legitimate at the level of a single transaction.
- Partner APIs inherit your trust and rarely your controls; rate limits, scoped credentials and per-partner anomaly baselines are the minimum.
- Velocity and behavioural controls catch what per-transaction rules cannot, but only if false-positive handling is designed with customer service.

CONTENTS

01 Where the losses actually come from

02 Social engineering of the customer

03 SIM swap and the limits of SMS

04 Agent networks and insider access

05 Partner APIs and the trust you lend out

06 Testing the things that actually break

01 Where the losses actually come from

Teams new to these platforms tend to invest first in the transport and the protocol – and those layers are, in our testing, generally the strongest part of the stack. The loss paths that matter run through people and process: a customer persuaded to authorise a transfer, a SIM re-issued to an attacker, an agent reversing transactions against their own float, an operations user with a query tool that can also write, a partner integration whose credentials were committed to a public repository.

Framing fraud as a product problem rather than a cryptography problem changes which controls get built, and it is the single most useful shift we see in teams that bring their loss rate down.

02 Social engineering of the customer

The most common pattern needs no technical compromise at all: the customer is contacted by someone presenting as the provider, is given a plausible reason for urgency, and authorises the transaction themselves. Every authentication control the platform holds is satisfied, because the legitimate user performed the action.

Controls that work here act on context rather than identity. A first-time recipient, an unusual hour, an amount outside the customer's pattern, a transaction initiated during an inbound call – none of these are individually conclusive, and in combination they are strong enough to justify a delay, a step-up confirmation, or an out-of-band check. Customer-facing friction is a cost, and the design question is where to spend it, not whether to.

03 SIM swap and the limits of SMS

Where a one-time passcode is delivered by SMS, control of the phone number is control of the account. SIM re-issue processes sit with mobile network operators rather than with the financial institution, and they are reachable by social engineering of customer-care staff and by insider collusion. This is a well-documented attack path, not a theoretical one.

The durable mitigations do not try to make SMS trustworthy. Binding a session to a device key or a provisioned app instance means a re-issued SIM alone does not grant access. Where available, operator SIM-change signals can gate high-risk actions for a cooling-off period after a swap. Where SMS remains the only channel, treat the number as an identifier rather than an authenticator and require a second, non-telephony factor for value transfer.

04 Agent networks and insider access

Agent fraud is hard to see transaction by transaction, because each transaction is individually legitimate. It shows up in patterns: reversal rates well outside the peer distribution, float movements

that do not match customer activity, clusters of registrations sharing a device or a location, concentrations of failed PIN attempts.

Insider risk follows the same shape. The control that reliably reduces it is separating read from write in operational tooling, with privileged write actions requiring a second approver and producing a log the actor cannot edit. Most insider cases we have investigated were possible because a support tool combined lookup and adjustment in one interface, used by a broad group, with logs held in the same system.

- Baseline each agent against peers, not against a global threshold.
- Separate read and write in support tooling; require dual approval for balance adjustments and reversals.
- Write privileged action logs to an append-only store outside the application.
- Review entitlements on a schedule, and immediately on role change.

05 Partner APIs and the trust you lend out

Every partner integration extends your trust boundary to an organisation whose controls you do not run. The recurring findings in our assessments are credentials scoped far wider than the partner's use case, no per-partner rate limiting, no anomaly baseline per credential, and no way to revoke one partner without an outage for all.

Treat each integration as its own tenant: narrowly scoped credentials, independently rotatable, independently rate-limited, with its own traffic baseline and its own kill switch. Add a contractual requirement for breach notification to you, within a stated window, and test that the contact route works before you need it.

06 Testing the things that actually break

Penetration testing that stops at the mobile application and the public API surface will report a short list of low-severity findings and miss the fraud paths entirely. The tests that find real exposure target business logic: transaction state machines under concurrent requests, authorisation checks on every step rather than the first, reversal and refund flows, USSD session handling, and the support tooling that staff use.

We also recommend testing the human processes in scope — with written authorisation — because the SIM-swap and customer-service paths are where the losses are, and no amount of code review surfaces them.

About this publication

This document is published by SCIAT AFRICA, a cybersecurity and digital forensics agency based in Nairobi, Kenya. It is free to read, print and circulate inside your organisation. The web edition, which carries the same content, is at <https://sciatafrica.com/research/mobile-money-fraud-controls>.

It is general guidance, not legal advice and not a substitute for an assessment of your own environment. Where it refers to a law, regulation or standard, read the instrument itself before relying on it – and talk to us if you want help applying it to your estate.

SCIAT AFRICA

info@sciatafrica.com · +254 794 946633 · <https://sciatafrica.com>