Ethereum Hackathon Proposal: GeoTrust

1. The Problem We're Solving

Online banking fraud is escalating due to weak geolocation-based authentication. Most fraud detection systems fail to consider physical movement feasibility and rely solely on IP data or heuristic models. This creates vulnerabilities when malicious users exploit VPNs, proxies, or spoofed identities to impersonate genuine users and execute unauthorized transactions.

2. Our Ethereum-Based Solution

GeoTrust is a decentralized fraud detection protocol built on Ethereum that leverages:

- IP geolocation,
- Google Maps Timeline GPS trails, and
- GIS-based travel feasibility

to verify a user's location before authorizing sensitive transactions.

Core Features:

- Compare current IP location with historical GPS movements.
- Estimate travel feasibility using GIS (Haversine + dwell time + velocity).
- Assign a fraud risk score based on behavioral and geographic inconsistency.
- Use soulbound tokens (SBTs) to flag high-risk accounts or verify trusted behavior.
- All risk evaluations are recorded on-chain for transparency and audit.

3. Planned Tech Stack

Smart Contracts: Solidity, ERC-721/5192 (SBT), Chainlink

Backend: Node.js, Express.js, Python (GIS calculations), IP reputation APIs

Frontend: React, Ethers.js, TailwindCSS

Data Sources: IP Geolocation APIs, Google Maps Timeline (user-permissioned), Device location

GIS Tools: Turf.js, Haversine formula, OpenStreetMap

Blockchain Network: Ethereum Sepolia/Testnet -> L2 (Polygon/Optimism)

Visualization: Leaflet.js, D3.js

DevOps: Hardhat, Infura, GitHub Actions, IPFS

4. Roles of Each Team Member

Tummuri Naga Veera Venkata Sai Ram (Blockchain & GIS Integration Lead):

- Solidity development, GIS computation model, smart contract deployment, IP/GPS logic.

Chittala Naga Venkata Jayadeep (Backend & Risk Engine Developer):

- API aggregation for IP reputation and GPS data, backend fraud score calculator, data normalization.

GitHub Profiles:

- Sai Ram: https://github.com/sai624183
- Jayadeep: https://github.com/jayadeepchittala

5. What Makes Our Solution Innovative or Impactful

- First-of-its-kind fusion of GIS with blockchain for fraud detection
- Dynamic travel feasibility using dwell time + velocity checks
- On-chain transparency for auditability of fraud decisions
- Soulbound token (SBT) system for persistent fraud flags or trust scores
- Decentralized yet privacy-preserving-users consent to GPS data usage

Our system bridges the physical and digital world, providing a far more accurate fraud detection mechanism than traditional rule-based or IP-only models. It empowers both users and institutions with real-time, explainable, and tamper-proof risk insights.

Comprehensive Approach to Fraud Detection Using IP Address and GIS Integration

1. Content

This document presents a strategy for detecting and preventing fraudulent online banking transactions by using IP geolocation data, GIS analysis, and Google Maps Timeline information. Before transaction, the algorithm will analyze the current IP address and the past timelines of other transactions where the past data is taken from google maps timeline and it is given as input to the GIS (geographic information system) , here the GIS will compare the data from both present IP address and the past timelines to check . If the locations didn't match, then the algorithm will use the dwell time at previous location and the estimated travel speed to the new location to check if it is possible to travel in the estimated time. If it is suspicious, it will notify the user and bank.

2. Detailed Description of Data Sets

Fraud detection in digital banking requires a multi-dimensional approach to location verification. The following data sets are used:

2.1 IP Geolocation Data:

- Captured at each login or transaction attempt.
- Fields: IP Address, Latitude, Longitude, City, Region, Country, ISP, Proxy/VPN status.
- Source: Real-time lookup using services like IP address, google maps timeline, GIS.
- Purpose: Determine user's apparent location from their internet connection.

-

2.2 Google Maps Timeline Data:

High-resolution, timestamped GPS data from the user's mobile device.

- Provides exact geographic coordinates and temporal context for user movements.
- Source: Synced via user's Google account with proper permissions.
- Purpose: Establish actual user movement patterns to validate IP-based locations.

2.3 Transaction Logs:

- Captures every financial activity within the bank's system.
- Fields: Transaction ID, Timestamp, User ID, IP Address, Device ID, Transaction Amount.
- Purpose: Link financial behavior with geolocation data and establish temporal sequences.

2.4 Device Location Data (Optional):

- Obtained in real time from device sensors.
- Fields: Latitude, Longitude, Timestamp, Device ID.
- Purpose: Additional verification of user's physical presence at the transaction time.

2.5 IP Reputation Databases:

- External lists that classify IPs as safe, risky, or malicious.
- Fields: IP Address, Risk Score, Proxy/VPN/Anonymizer Flags, Blacklist Status.
- Purpose: Detect and block access attempts from known fraudulent sources.

3. Solution Approach

The fraud detection mechanism combines spatial and behavioral analytics to determine the authenticity of a user's location.

-

3.1 Data Collection and Preprocessing:

- Log all user sessions with IP addresses and timestamps.

Query geolocation services to convert IP addresses to coordinates.

- Retrieve the latest GPS data from Google Maps Timeline for comparison.
- Store and timestamp all historical location and device records for each user.

3.2 Geographic Consistency Checks:

- Calculate the distance between the new IP location and the last GPS location.
- Use the Haversine formula for distance calculation.
- Determine dwell time at previous location and time elapsed since last verified GPS location.- Compute estimated travel speed and compare with realistic human capabilities (e.g., >900 km/h is suspicious).

3.3 Risk Scoring Engine:

- Assign a fraud risk score using weighted rules:
- Travel speed exceeds normal limits.
- IP and GPS locations conflict.
- VPN/proxy usage is detected.
- Device fingerprint mismatch or new device.
- Apply threshold-based decisions:
- Score 70-89: Notify user for verification.
- Score 90: Block transaction and alert fraud response team.

3.4 Fraud Classification and Notification:

- If a mismatch is confirmed and risk score is high:

-

- Mark the transaction as potentially fraudulent.

- Send alerts to the user and bank security system.

- Lock the account temporarily if auto-block is enabled.

Require user to confirm recent activity via MFA or call center.

3.5 GIS Visualization and Monitoring:

- Map historical and real-time movement on GIS platforms (e.g., ArcGIS, QGIS, Leaflet.js).

- Display:

- GPS trails from Google Timeline.

- IP geolocation points.

- Suspicious transition paths with speed markers.

- Enable analysts to review fraud trails visually for deeper investigation.

3.6 Continuous Learning and Optimization:

- Use confirmed fraud cases to retrain models and update scoring rules.

- Incorporate feedback from manual investigations into algorithm adjustments.

- Update IP reputation sources regularly.

4. Contact details:

NAME: Tummuri Naga Veera Venkata Sai Ram

Gmail: tummurisairam@gmail.com

Github: https://github.com/sai624183

NAME: Chittala Naga Venkata Jayadeep

Gmail: nagavenkatajayadeepchittala@gmail.com

Github: https://github.com/jayadeepchittala