Security Review Report: ChainCrib

Overview

This document provides a security review of the Cardano integration for ChainCrib. This protocol uses NMKR to mint and manage NFTs representing fractional ownership of properties, and Blockfrost to submit transactions.

Endpoint Descriptions

POST /property

- **Purpose**: Create a new property.
- **Functionality**:
  - Calls NMKR to upload files related to property (i.e., images) and also obtains `nftUid` from response
  - Stores the property in the internal model with `nftId: nftUid`.

GET /property

- **Purpose**: Retrieve a list of properties.
- **Functionality**:
  - Fetches all properties from the application model.
  - No authentication required.

GET /property/user

- **Purpose**: Fetch properties associated with the authenticated user.
- **Functionality**:
  - Requires user authentication.
  - Retrieves user information and the fractions of properties they own.

GET /property/id

- **Purpose**: Fetch a specific property and ownership details.
- **Functionality**:
  - Requires user authentication.
  - Returns the property details and fraction owned by the authenticated user.

POST /property/buy

- **Purpose**: Buy fractions of a property.
- **Functionality**:
  - Requires user authentication.
  - Queries the property from the model.
  - Validates availability of fractions.

  - Accepts a user-generated signed transaction (`signedtx`) to transfer ADA for the purchase.
  - Submits the transaction to Cardano via Blockfrost.
  - Mints corresponding NFTs on NMKR using the associated `nftId`.
  - Sends NFTs to supplied address (`receiverAddress`)
  - Updates internal models (property, user, transaction_history).

GET /cardano/user/me

- **Purpose**: Retrieve user profile and property ownership.
- **Functionality**:
  - Requires authentication.
  - Returns user details and owned property fractions.

GET /cardano/transactions/user

- **Purpose**: Retrieve blockchain transactions related to the authenticated user.
- **Functionality**:
  - Requires authentication.
  - Queries and returns relevant Cardano transactions.

Security Considerations and Potential Vulnerabilities

These are some of the potential areas where vulnerabilities could be found. This list is not meant to be exhaustive, but only provides a starting point to investigate further.

Signed Transaction Handling

- **Potential Issues**:
  - Lacks validation of:
    * Correct amount (ADA) transferred.
    * Accuracy of the recipient wallet address.
    * Transaction uniqueness (risk of replay attacks).

External Service Dependency (Blockfrost)

- **Potential Issues**:
  - Dependency on Blockfrost for transaction submission and status checking.
- **Questions to Clarify**:
  - What are exactly the guarantees offered by Blockfrost and the submit endpoint?
  - When can the transaction be considered final/confirmed on-chain?

External Service Dependency (NMKR)

- **Issues**:
  - Application depends on NMKR being available for creating new properties and buying fractions.
  - The application's NMKR account must be sufficiently funded for all operations.

NFT Identifier (`nftId`) Integrity

- **Potential Issues**:
  - If `nftId` is modified in the application model, it could lead to users buying property fractions for the wrong price.