A diagram of a diagram

Description automatically generated with medium confidence

Overall Data Flow:

1. The client initiates a transaction. The client’s request is sent to relevant backend service, which is responsible for handling the client’s request.
2. Inside the backend service, necessary validations will be performed:
   1. User authentication
   2. Input validation.
   3. Business logic checks

Once validated, the backend service formats the transaction data as required by the Transaction Broadcaster Service. (TBS)

1. The backend service makes an API request to the TBS, specifically to the ‘/transaction/broadcast’ endpoint. The request includes the transaction data and the needed metadata.
2. The TBS API gateway receives the request then forwards it to the Request Handler.
3. Internal TBS Processing:
   1. Request Handler validates the request and extracts the transaction data, then forwards it to the Transaction Signer.
   2. Transaction Signer signs the transaction with the appropriate cryptographic keys managed by the TBS.
   3. The signed transaction is placed in a queue until it can be broadcasted to the blockchain network.
4. Blockchain Interaction:
   1. Blockchain Interface: The TBS sends the transaction to the blockchain network via an RPC call to an Ethereum node.
   2. Broadcast Confirmation: The Ethereum node accepts the transaction and broadcasts it to the network.
5. Status Tracking and Reporting:
   1. Status Tracker: Monitors the status of the transaction on the blockchain network. Once the transaction is confirmed or fails, the status tracker updates the record in a persistent storage.
   2. Admin Interface: Provides a real-time view of the transaction status. If the transaction fails, an auto retry mechanism can be implemented, or an admin can manually retry the broadcast.
6. Response to Client:
   1. Status Update: The TBS updates the backend service regarding the status of the transaction.
   2. Client Notification: The backend service then communicates the result of the transaction back to the client.

**NOTE:**

* Retry Mechanism:
  + Failed Transaction Handling: If a transaction fails to broadcast or doesn't get confirmed, the TBS's retry logic will attempt to rebroadcast the transaction based on pre-defined rules.
  + Admin Intervention: In case of persistent issues, an admin can use the Admin Interface to investigate and manually trigger a retry.
* Role of Status Tracker:
  + Reliability: ensures that transactions are reliably processed by keeping track of each transaction’s confirmation status on the blockchain.
  + Operational control: provides necessary information for Admin interface, which can be crucial for manual intervention and management.
  + Debugging: provides visibility into the state of transactions for debugging purposes.