GreenRoulette: A Decentralized Betting Platform

1 Introduction

GreenRoulette is a decentralized application (dApp) built on the Ethereum blockchain that allows users to bet in a game of chance while contributing to charitable causes. The dApp is powered by a smart contract with a central pool of Ether (ETH). When players lose a bet, their funds are added to this pool. The pool's growth benefits both charitable organizations and partners who stake their ETH in the platform.

2 Pool Distribution

Every month, a fixed percentage of the total pool is distributed as follows:

- **4%** of the pool goes to various **charities**.
- **1%** of the pool goes to the **house**.
- **1%** of the pool is distributed to **partners**.

Partners are users who have staked their ETH in the GreenRoulette pool. The amount each partner earns is proportional to the amount of ETH they have staked compared to the total amount of ETH staked by all partners.

3 Partner Earnings Calculation

To illustrate how partner earnings are calculated, consider the following example:

- Suppose a user has staked 1 ETH and another partner has also staked 1 ETH.
- The total amount staked by all partners is 2 ETH.
- Each partner holds 50% of the total staked amount.
- If the pool size is 100 ETH, 1% of the pool, which is 1 ETH, is distributed to the partners.
- Each partner would receive 0.5 ETH for that month.

4 Monthly Fund Distribution

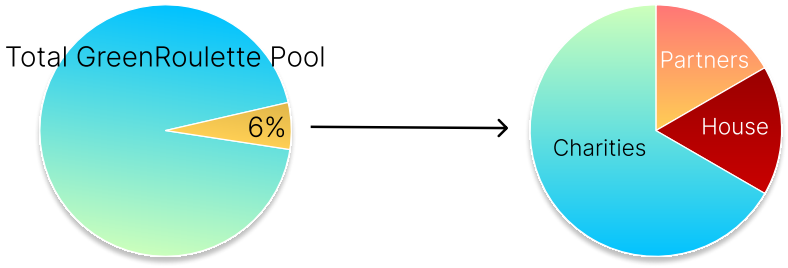

Figure 1: Distribution of Funds Every Month

How the funds are distributed every month is illustrated in Figure 1.

5 Random Number Generation

GreenRoulette utilizes Flare's FTSO (Flare Time Series Oracle) mechanism for generating secure random numbers to ensure fairness in game outcomes. The random number generation is based on a secure commit-reveal scheme and involves the following phases:

5.1 Commit Phase

During the Commit phase, each data provider generates a random number locally and prepares their submissions for each data feed. They then encode this data into a 4-byte vector and publish a hash commitment on-chain. The commitment is created as follows:

$$\text{Hash(address, voting\_epoch\_id, random\_number, price\_data)}$$

The locally generated random number serves as a blinding factor to protect against search attacks.

5.2 Reveal Phase

In the Reveal phase, data providers reveal all inputs to their hash commitments. This includes the locally produced random numbers, which become available on-chain. The revealed values must match the committed hashes, ensuring integrity.

5.3 Random Number Calculation

For each voting epoch (90 seconds), an overall random number R is generated from the local random numbers r_i produced by each data provider. This is done using the formula:

$$R = \left(\sum_i r_i \right) \mod N$$

where $N = 2^n$ denotes the maximum possible size of the individual n-bit random numbers. This mechanism ensures that R is a uniformly generated random number, provided at least one input is an honestly generated uniformly random number.

5.4 Security Mechanism

The protocol includes a security mechanism to prevent withholding attacks. If any data provider omits a reveal or if the revealed data does not match the commitment, the random number for that epoch is marked as insecure, and the data provider is penalized. This encourages honest participation and ensures that the generated random number is fair and unpredictable.

6 Conclusion

GreenRoulette provides an innovative way to participate in a game of chance while supporting charitable causes. By staking ETH in the platform, users can become partners and earn a share of the monthly distribution proportional to their staked amount. The use of Flare's FTSO for secure random number generation further enhances the fairness and security of the platform, ensuring trustworthy and transparent outcomes for all participants.