去中心化红包

1、什么是去中心化红包

传统红包的发放，比如 QQ，微信等，底层是由一个抽奖系统实现用户金额的发放比例的，这种模式下的红包发放是完全中心化的。

去中心化红包，就是将去中心化随机数应用到红包发放的场景中，实现红包的真随机分发，保证每个用户可以公平地收到相应的金额。它和传统的红包发放流程不同，这里的随机数在区块链上生成，是公开可见、完全去中心化的。

去中心化红包和传统的红包发放方式最大的不同就是公平性。去中心化红包使用的是去中心化随机数，是由多方共同参与生成的，同时还实现了随机数的公开可验证，相比传统红包的集中式服务端伪随机数生成方式更加公平。

2、去中心化红包的论文基础和已有方案

去中心化红包的核心在于去中心化和可验证随机数的生成，可验证随机数的概念和理论原型由 Micali 提出，主要思想是对伪随机数生成函数进行扩展从而增加可验证性与不可预测性。在 Micali 的研究中参与者能够使用伪随机公式 v=fs(x)在定义域任一点的自变量 x 与证据 Proof，检验 fs(x)输出 v 的随机性和不可预测性，这种数学关系式称为可验证随机函数（Verifiable Random Function，VRF），这种 VRF 的可验证随机函数为参与者提供了很高的参与信任感，可以保证参与者的有效参与和自证明。

现阶段对于去中心化随机数的生成和应用已经有了较多的研究，在 Mingxiao Du 团队的论文《A Blockchain-based Random Number Generation Algorithm and the Application in Blockchain Games》中给出了基于区块链游戏的随机数生成方案：

1.Game provider 生成一个随机数 Np 和 public-private key pair，game provider 使用公钥加密随机数为 E(Np)，将信息 massage:{ E(Np) & GameID } 发送给区块链

2.区块链使用 smart contract 检测 GameID，如果为新游戏，则记录相应信息 record:{ txid1 & E(Np) & GameID }，并将 txid1 返回给 Game provider

3.Game provider 发送 txid1 给所有参与者，参与者根据 txid1 自行检测信息。参与者 i 将自己的随机数发送给 Game provider massage:{ GameID & Ni }，Game provider 发送收集到的随机数给区块链合约 record:{ E(Np) & GameID & (N1...Ni...Nn) }

4.Smart contract 检测信息，生成记录 record:{txid2 & txid1 & E(Np) & GameID & (N1...Ni...Nn) }，返回 txid2 给 Game provider

5.Game provider 广播 txid2 给所有参与者。所有参与者同意后可以开始生成随机数，参数包括 block_txid2_hash、txid2、N(p)、(N1...Ni...Nn)， 使用 f(x) 生成随机数 k

6.使用 k 进行游戏， 结束后 game provider 上传 record 和游戏结果 result 给智能合约

7.Smart contract 生成交易信息 massage:{ txid3 & txid2 & txid1 & E(Np) & GameID & (N1...Ni...Nn) & result & Operation & Private Key }

8.参与者使用私钥和 txid3 检测所有信息真实性

可以看到，在这个方案中有三个明显的缺陷，第一是交互周期过长，第二是随机数的传

递采用明文传输，第三是依然依赖于中心 Game provider 提供的随机数 Np，但这个方案在一定程度上提供了去中心化思路。

基于 Ethereum 的 randao 项目是一个相对简单和具有效率的去中心化随机数方案，他的基本实现思路如下：

1.收集参与者提供的随机数加密散列值，在某时间区间内收集加密散列以后期验证，为保证参与者完成整个随机过程，需要提供部分 gas 作为质押与散列一同发送给合约账户

2.将成功收集到的散列值提供者规定为最终随机数生成参与者，随后参与者将随机数发送至合约账户，合约账户验证所有接受到的随机数作为有效随机数，未提供随机数的用户不会终止随机数生成过程，质押不会返还。

3. 将收集到的随机数作为种子生成最终随机，完成最终随机后可将所有质押金额均分给参与用户

在 randao 中，通过先发送散列后发送随机的方式，解决了明文传输过程的安全问题，也通过 gas 抵押防止了一定的参与者作弊，但整体效率预设需要长达 6 个区块时间，交互过程较多，并不便于用户的使用。

3.本文的去中心化红包方案

为了适用于去中心化的环境，本文提出了基于可验证随机函数的去中中心化红包方案。为了使得红包分配过程中生成的随机数是公开可验证的，在本方案中使用了可验证随机函数（Verifiable Random Function,VRF）。可验证随机函数是一种将输入映射为可验证的伪随机输出的加密方案。首先，VRF 所得是一个随机数，其次由于包含生成者的私钥签名，验证者可以通过公钥确定随机数的合法性。

可验证随机函数的算法流程如下：

- 1.证明者生成一对秘钥，PK 和 SK；
- 2.证明者计算 result = VRF_HASH(SK,info)；
- 3.证明者计算 proof = VRF_Proof(SK,info);
- 4.证明者把 result 和 proof 递交给验证者;
- 5.验证者计算 result = VRF_P2H(proof)是否成立，若成立，继续，否则中止；
- 6.证明者把 PK，info 递交给验证者；
- 7.验证者计算 True/False = VRF_Verify(PK,info,proof),True 表示验证通过，False 表示验证未通过。

红包分配的实质是去中心化随机数的生成，本文的方案能够很好的解决随机数生成过程中的用户作弊行为，每名用户参与红包分配的过程都需要通过可验证延时函数生成对应的零知识证明，从而使得任何人都可以对用户的子份额生成过程进行验证，保证了最终随机数生成的公平性和可验证性。

本文的去中心化具体流程如下:

①发送人在客户端发送红包

②接收人在客户端选择是否点击领取，若不领取则不参与红包的分发

③接收人选择自己的随机数作为输入，通过 VRF 函数得到随机数子份额和零知识证明

④验证合约验证证明的有效性，若无效则出局，有效则将子份额信息存放到区块中。

⑤等待所有接收人将随机数子份额发送且被验证，默认 6 个区块的生成时间。

⑥合约将所有的随机数子份额作为 VRF 的输入，输出得到最终的随机数和零知识证明

⑦将最终生成的随机数作为种子为接收者分配红包的金额

⑧合约根据分配的红包金额进行交易转账
⑨转账成功，在客户端通知接收者。