Moonbeam

Moonbeam开发进阶课程

随机数预编译

PureStake开发者关系团队

课程导航

VRF原理概述
- 什么是VRF？
- 随机性
- 使用场景

Moonbeam VRF 架构

Moonbeam VRF 预编译接口

VRF预编译调用示例

VRF原理概述

什么是VRF？

- VRF（Verifiable Random Function）是可验证随机函数的缩写
- VRF是一种利用公钥加密算法的伪随机函数，可提供其输出值正确性的证明
- 密钥的所有者可以计算函数输出值以及任何输入值的相关证明
- 概念在1995年由MIT教授Silvio Micali（Algorand创始人）提出

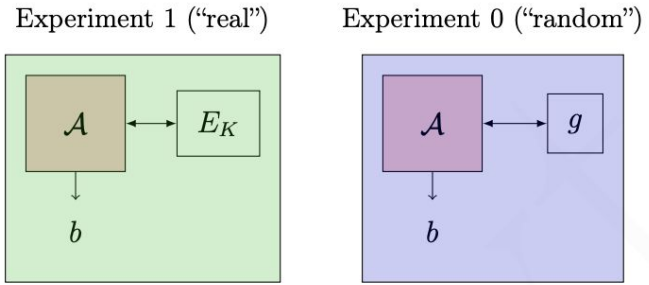
Input:（公钥，私钥，随机种子） → VRF函数 → *Output: (随机数，证明)*

随机性和伪随机性

- **随机性**可以定义为：

$$\forall f \in \mathcal{F}(l, L) \qquad f : \{0,1\}^l \mapsto \{0,1\}^L \qquad g(\cdot) \xleftarrow{\$} \mathcal{F}(l, L)$$

- 这种理论上的随机性是不存在于计算机中的，所以我们需要定义伪随机性
- **伪随机函数**可以定义为一个攻击者无法可靠的分辨一个伪随机函数和理论随机函数的输出：

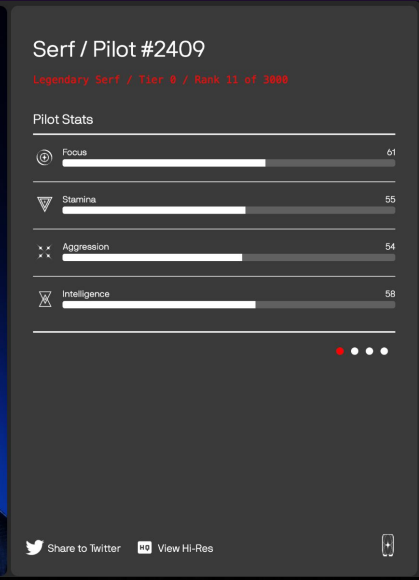


为什么伪随机函数的输出需要证明？

- 如果一个攻击者可以操控或影响一个“伪随机”函数的输出，他可以将很多“随机”的结果变为对他有利的结果，达成一种攻击向量
- 很多区块链和其它计算机系统的安全性依赖于伪随机函数的安全性
- 举例：验证人选择

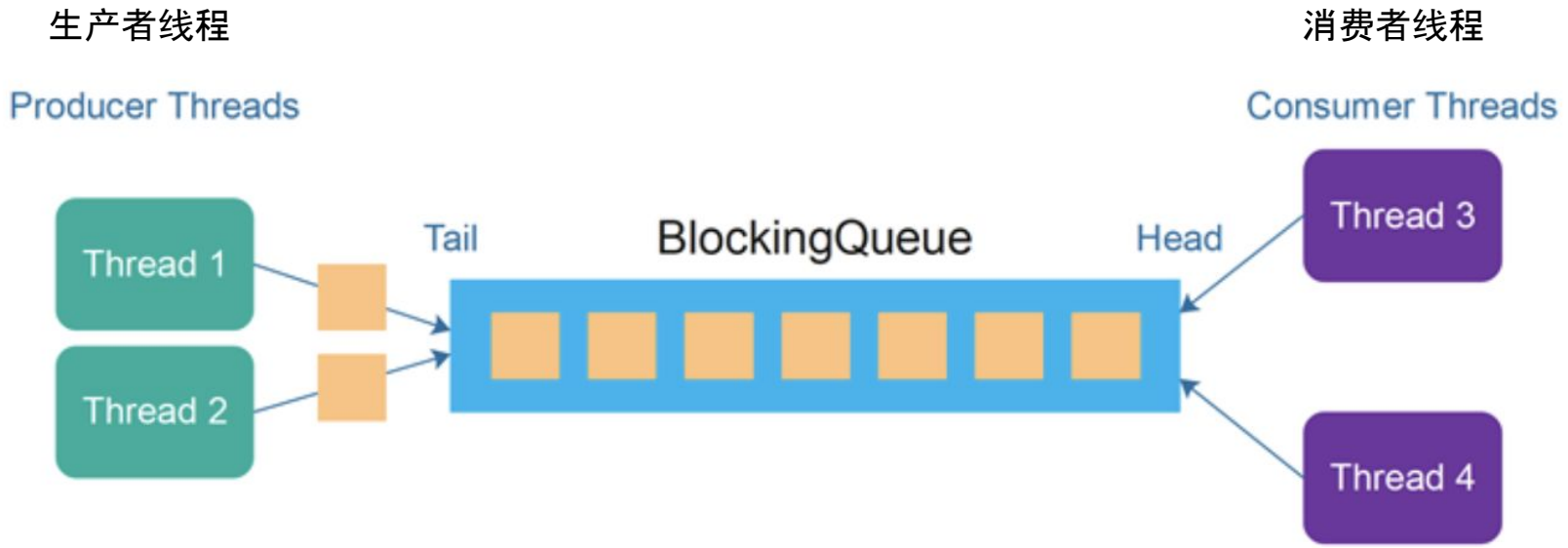
VRF使用场景

- 区块链游戏
- NFT铸造
- 共识算法选择验证人
- 零知识证明(可重置类)
- 波卡插槽蜡烛拍卖

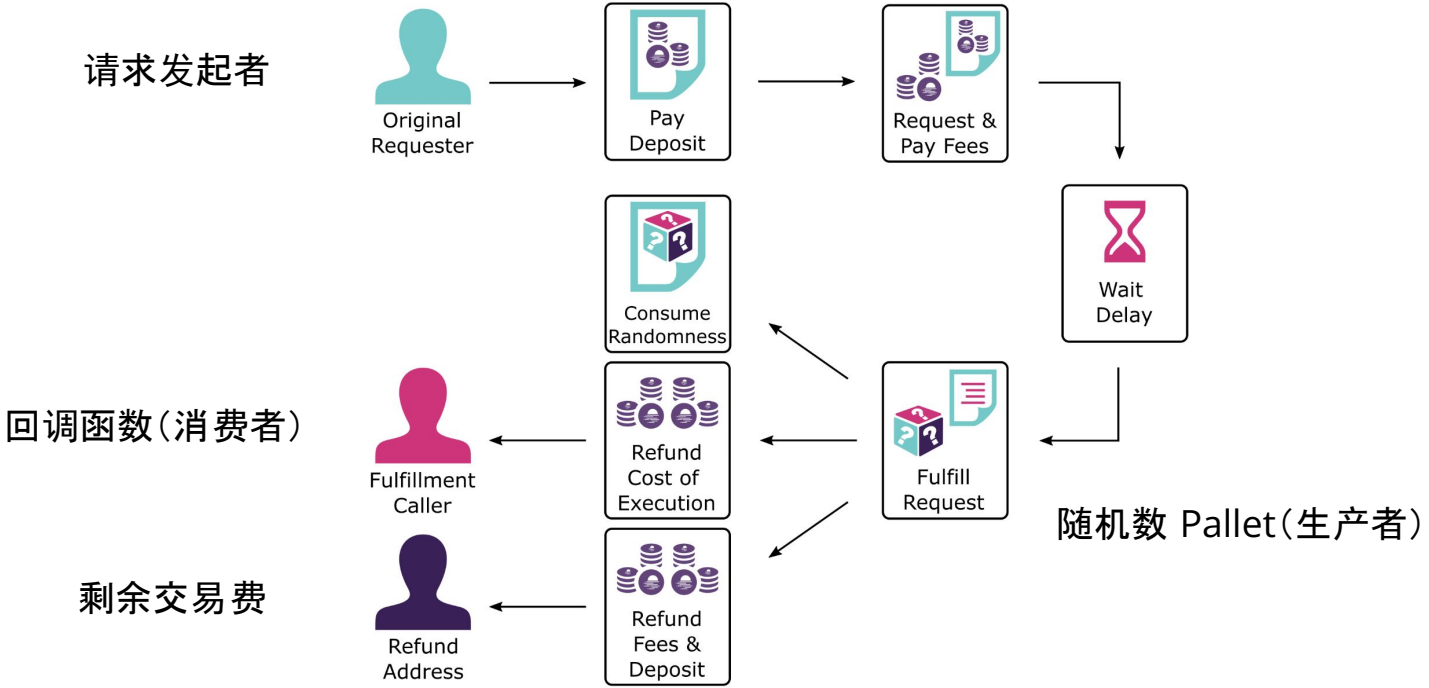


Moonbeam VRF 架构

生产者/消费者设计模式


生产者线程
消费者线程
Producer Threads
Consumer Threads
Thread 1
Thread 2
Tail
BlockingQueue
Head
Thread 3
Thread 4

Moonbeam VRF请求模型



Moonbeam VRF预编译接口

VRF预编译使用方法概述

- 需要使用VRF随机数输出的合约需要继承RandomnessConsumer接口，并实现fulfillRandomWords虚拟方法
- 通过Randomness预编辑发起生产请求并绑定一定数额的担保和交易费用；每一个请求需要绑定一份担保金
- 请求完成后，担保金和剩余的交易费会返回指定的退款钱包
- VRF预编译地址：`0x0000000000000000000000000000000000000809`

Variable	Value
MAX_RANDOM_WORDS	100 words
MIN_VRF_BLOCKS_DELAY	2 blocks
MAX_VRF_BLOCKS_DELAY	2000 blocks
REQUEST_DEPOSIT_AMOUNT	1 DEV

Randomness.sol

方法：

- **requestLocalVRFRandomWords**(*address* refundAddress, *uint256* fee, *uint64* gasLimit, *bytes32* salt, *uint8* numWords, *uint64* delay) — 请求从平行链VRF生成的随机词；随机词为uint256数组
- **requestRelayBabeEpochRandomWords**(*address* refundAddress, *uint256* fee, *uint64* gasLimit, *bytes32* salt, *uint8* numWords) — 请求从中继链BABE共识生成的随机词；随机词为uint256数组
- **fulfillRequest**(*uint256* requestId) — 完成请求并调用消费者合约的回调函数fulfillRandomWords。如果请求可以完成，则将退还剩余的费用

事件：

- **FulfillmentSucceeded**() - 当请求成功执行时发出
- **FulfillmentFailed**() - 当请求未能执行履行时发出

RandomnessConsumer.sol

方法：

- **fulfillRandomWords**(*uint256* requestId, *uint256[] memory* randomWords) - 处理给定请求的 VRF 响应。此方法由调用 rawFulfillRandomWords 触发
- **rawFulfillRandomWords**(*uint256* requestId, *uint256[] memory* randomWords) - 在调用随机预编译的fulfillRequest 函数时执行。验证调用的来源，确保随机性预编译是来源，然后调用fulfillRandomWords方法

VRF接口合约链接

https://github.com/PureStake/moonbeam/blob/master/precompiles/randomness/Randomness.sol

https://github.com/PureStake/moonbeam/blob/master/precompiles/randomness/RandomnessConsumer.sol

VRF预编译调用示范

RandomnessDemo.sol合约链接

https://github.com/PureStake/moonbuilders-academy/blob/main/chinese/advanced-course/week5-VRF-precompile/RandomnessDemo.sol