1. 优化过程

观察1：写入合约的全局变量会消耗大量的gas值。

优化方案：

1. 中间过程尽量使用局部变量进行，仅当需要记录数据时才存入全局变量。【节约158万（->41万）gas和一些时间】
2. 只有通过验证的state才会进入beststate，因此beststate中无需存储签名。【节约85万（->11万）gas和大量时间】
3. 在上传地址时使用局部变量作为累加器，最后再赋值给全局变量，而不是每次都累加在全局变量上。【节约2万gas】
4. 将总人数等不会改变的变量保存到局部变量，避免重复多次读取同一个全局变量浪费gas值。【节约若干gas】

观察2：函数的定义和调用都会额外消耗gas值。

优化方案：

1. 将所有验证内容写在一个函数中。（但降低了代码的可读性）【节约几万gas，部署时节约更多】
2. 原先上传地址和state需要多次上传，现在改为一次性上传。（需要进行编码和解码）【节约69万gas】
3. 删除用于调试的函数。【部署时节约若干gas】
4. 一些小问题

在solidity中，局部变量基本不消耗gas值，但由于以太坊模拟器的栈大小存在限制，因此一旦局部变量数量太多或者表达式太复杂，编译器就会报错（但对局部变量的大小似乎无限制）。为此，需要减少变量的数量，主要采用了一下几种方法：

1. 仅在需要变量的作用域定义局部变量，这样出了作用域以后这些局部变量就可以释放。
2. 将多个变量捆绑进一个变量中，需要使用时利用位运算截取不同的位得到所需的值。(但降低了代码的可读性)

目前在验证state是否正确的函数中，AuditSin刚好达到局部变量的数量上限， AuditDou也接近上限，将来可能无法继续扩展。

1. gas值

一次性费用：208万

合约部署：176万

上传地址：32万

检验费用：114万->76万

验证：57万（其中验证签名42万）

更新beststate：57万->19万