Smart Contract Audit

Realitio

By,
Shebin John

7th July 2020

Index

Note: Some threat levels or headings might be empty if there is no vulnerability/updates/suggestions found.

Introduction

The contract audited here is/are:
https://github.com/realitio/realitio-contracts/blob/c64f28c/truffle/contracts/Realitio.sol
https://github.com/realitio/realitio-contracts/blob/3cc6c14/truffle/contracts/RealitioERC20.sol
https://github.com/kleros/kleros-interaction/blob/142f9dd/contracts/standard/proxy/RealitioArbitratorProxy.sol
and was shared by Kleros for auditing purposes while working with them.

Based on the audit, we were able to find no High threats, no Medium threat, and no Low threat with some other changes in the optimization, readability, typos, and comments section.

High Threats

1. None

Medium Threats

1. None

Low Threats

1. None

Optimization & Readability

1. `require` to use instead of assert in RealitioSafeMath32 in line 12. Similar problem in RealitioSafeMath256 as well at Line 13, 25 & 31
2. `templates` could be a mapping from uint256 to bool for better gas optimization.
3. There is actually no apparent use of askQuestion() I believe, except that it does not take the number of tokens parameter. Which could be filled zero (in front end) automatically if there are no question bounty being declared and no question fee for the arbitrator.

Typo's & Comments

1. Comments not written for `_askQuestion()`, `_addAnswerToHistory()`, '_updateCurrentAnswer()', `_payPayee()` and `_processHistoryItem()`
2. Better naming of local variables or assisting comments for those variables in function `claimMultipleAndWithdrawBalance()`
3. L321 to L323 needs correction in comments and L323 parameter is commitment_id instead of commitment
4. It should be specified that the payment is in tokens for the arbitration question fee.

Suggestions

1. Here, the person who asks the question has the right to put the arbitrator address as well. Which can be considered decentralized, but also dangerous if the person who might be answering it could be challenged and won against, even if the answer was right. Either we might use a TCR/General List in the contract of trusted arbitrators, and ask the person who asks the question to use one among the arbitrator from the list. Anyone should be allowed to add an arbitrator to the list as well (Kleros can handle that department well.)
2. At the moment, RealitioERC20 is only compatible with a single token. So, for each token a new contract has to be deployed. Something like ERC1155 would have helped to handle multiple tokens in a single contract.