SMART CONTRACTS EXPERTISE — RIGHT WAY TO SUCCESS
The Abyss DAICO Smart Contract Audit

If you have any questions concerning smart contract design and audit, feel free to contact zoia@oceanico.io

Content

Description of the complex of procedures for auditing a smart contract

1. Primary architecture review

- Checking the architecture of the contract.
- Correctness of the code.
- Check for linearity, shortness and self-documentation.
- Static verification and code analysis for validity and the presence of syntactic errors.

2. Comparison of requirements and implementation

- Checking the code of the smart contract for compliance with the requirements of the customer code logic, writing algorithms, matching the initial constant values.
- Identification of potential vulnerabilities

Terms of Reference for the creation of a smart contract

The Abyss DAICO Smart Contract

TOKEN SALE DETAILS:

- Token symbol: ABYSS
- Decimals: 18
- TOKEN SALE (DAICO):
- Sale date: MAR 7 - APR 7, 2018 (09:00 UTC)
- Rate: 1 ABYSS = 0.24 USD

(We accept ETH and BNB. We will set a course for ETH and BNB in a few days before the start of Token Sale, depending on the market rates)

- Soft cap: 6.000.000 USD (If this goal is not met, all funds will be returned)
- Max total token supply: 603.750.000 ABYSS (All unsold tokens will be burnt by Smart Contract)

INTERNATIONAL

- Hard cap: 40.000.000 USD

Minimum contribution is 0.1 ETH
Days 1-7 Maximum contribution is 10 ETH
Days 8+ no ETH contribution limit

- 300.000 BNB

Minimum contribution is 1000 BNB tokens
Acceptable from the international (non U.S.) participants only

UNITED STATES

- Hard cap: 20.000.000 USD
- Minimum contribution is 1 ETH
- Days 1-7 Maximum contribution is 100 ETH
- Days 8+ no ETH contribution limit

Additional information about the smart contract was taken from the sources below

- https://www.theabyss.com/
- https://medium.com/theabyss/the-abyss-smart-contract-redeveloped-in-compliance-with-daico-concept-2d22415f9804

List of audited files

https://github.com/theabyssportal/DAICO-Smart-Contract

The following 21 .sol files were audited:

- AbyssToken.sol
- BufferVoting.sol
- Crowdsale.sol
- Fund.sol
- OracleVoting.sol
- Pausable.sol
- RefundVoting.sol
- TapVoting.sol
- VotingManagedFund.sol
- ICrowdsaleFund.sol
- IVotingManagedFund.sol
- SafeMath.sol
- MultiOwnable.sol
- Ownable.sol
- ERC20Token.sol
- IERC20Token.sol
- ITokenEventListener.sol
- LockedTokens.sol
- ManagedToken.sol
- TransferLimitedToken.sol
- BaseVoting.sol

Review of smart contract #1

The Abyss DAICO smart contract review #1

https://github.com/theabyssportal/DAICO-Smart-Contract

Important

1. Possible bug

1.1. BaseVoting (54): require(_startTime > now && _endTime > startTime);

1.1.1. _endTime > startTime always true, because startTime is state variable and it zero by default.

Code quality

2. Overview

2.1. Code quality is very good. Code style follow to the solidity recommendations. Almost all public methods have comments.

3. Critical

3.1. No critical issues found;

4. Gas usage

4.1. LockedTokens.Tokens: field lockEndTime might be stored as uint64. It saves about 1k gas;

4.2. bonusWindow#EndTime might be constants, it saves constructor gas; and the same thing with any TokenWallets;

4.3. Mappings telegramMembers and telegramMemberHadPayment might be merged to one mapping with uint (e.g. 1 means registered, 2 means had payed) value instead of bool.

5. General

5.1. External specification uses when it is not required: it does better performance when memory holded object passed as parameters;

5.2. Using modifiers for single invocation looks excessively, and it make code more unclear;

5.3. It's better to avoid using global properties (like msg.sender, msg.value) inside internal method, and pass values as arguments;

5.4. There is a duplicate logic implementation in processBNBContribution and processContribution which might be shared;

6. Notice

6.1. There is no possibility to transfer tokens to the locked account;

6.2. When constant defined as public, compiler generates accessor method for it;

6.3. If any hard cap will almost reached user must enter right value (calculated by themselves);

6.4. TokenPrice might be set only once (per crowdsale contract) - there is no chance to fix it; also, token price might be 0;

6.5. BNB contributions is not included to the soft cap checking.

Results of contract audit

https://github.com/theabyssportal/DAICO-Smart-Contract

The information in this report is a list of recommendations what needs to be done to ensure the quality and security of the smart contract.

The OCEANICO experts conducted the verification of the smart contract. Based on the results, the customer's developers were given recommendations for optimizing the smart contract code.

This smart contract complies with the specifications specified in the terms of reference.

During the audit of the contract, critical errors and possible vulnerabilities were not identified.

If changes are made to the functionality of the contract, please submit the smart contract for re-examination to the OCEANICO experts (zoia@oceanico.io).