OKC Treasury

Contract Audit Report

VER 1.0

29 March 2022

No. 2022032914481

Project Summary

1. Project Introduction

OKC Treasury is proposal governance and reward issuance contract on the OKC chain. The contract has two types of proposals: periodic reward proposals (each cycle to a specified address, reward a set amount of OKT) and fixed reward proposals (application to a specified address, fixed reward a limited amount of OKT), when the user has a certain number of OKT can send proposals, other users vote before the deadline, the administrator can count the number of valid votes by clean, count after the deadline, after judging whether the proposal is successful, call execute to execute the proposal.

2. Audit Summary

Project Name	OKC Treasury	**Platform**	N/A
Token	N/A	**Token symbol**	N/A
Start date	22 March 2022	**Language**	Solidity
End date	29 March 2022	**Website**	N/A
Github	https://github.com/okex/OEC-Treasury/tree/2bdc1a54e22285cc61ae2a82e45bd52a9fc0be80	**whitepaper**	N/A

3. Audit Scope

ID	File	SHA-256 checksum
contracts	Governance.sol	2296e959513da671e6bbd06b911fff0a530c7d0893e53f25fe78882b34ff081f
contracts	Treasury	d4df8420643a2ef73180e1506988eca520529267bc3b5969b1505339f150c6fb

4. Code Structure

```
├─── contracts
|    ├─── Treasury.sol
|    └─── Governance.sol
```

Audit Report Summary

1. Audit Methods

The audit was conducted to gain a clear understanding of how the project was implemented and how it works. The audit team conducted in-depth research, analysis, and testing of the project code and collected detailed data. In this report, the audit team will list in detail each issue identified, where it is located, the root cause of the issue, and a description of the issue, and will recommend changes to the issue accordingly.

Audit methods	Static analysis, Manual Review

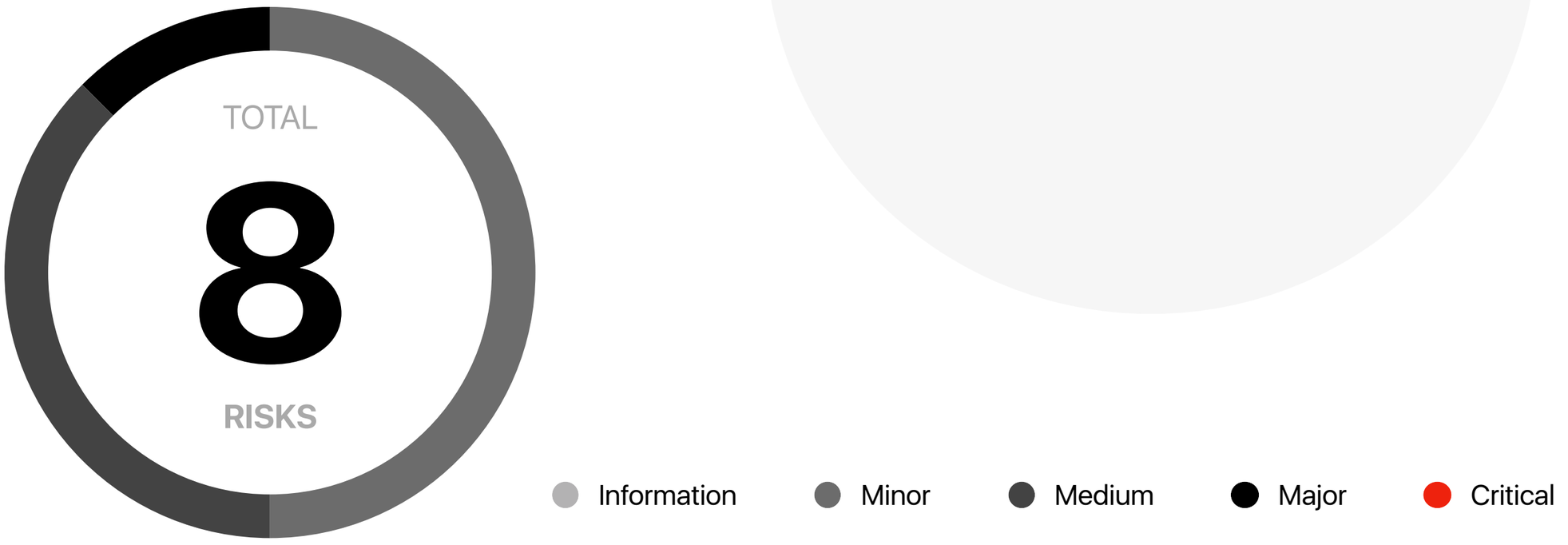
2. Audit Process

Steps	Operation	Description
1	Background	Read project descriptions, white papers, contract source code, and other relevant information the project team provides to ensure a proper understanding of project functions.
2	Automated testing	Scanning source code mainly with automated tools to find common potential vulnerabilities.
3	Manual reveiw	Engineers read the code line by line to find potential vulnerabilities.
4	Logical proofread	The engineer will compare the understanding of the code with the information provided by the project and check whether the code implementation is in line with the project white paper information.
5	Test case	Including test case design, test scope analysis, symbolic execution, etc.
6	Optimization items	Review of projects in terms of maintainability, safety, and operability based on application scenarios, deployment methods, and latest research results.

3. Risk Levels

Risk level	Description
Critical	Fatal risks and hazards that need to fixed immediately.
Major	Some high risks and hazards that will lead to related problems that must be solved
Medium	Some moderate risks and pitfalls may lead to potential risks that will eventually need to be addressed
Minor	There are low risks and hazards, mainly details of various types of mishandling or warning messages, which can be set aside for the time being
Information	Some parts can be optimized, such problems can be shelved, but it is recommended that the final solution

4. Audit Results



ID	Audit project	Risk level	Status
1	Reentrancy	None	
2	Injection	None	
3	Authentication bypass	None	
4	MEV Possibility	Major	Acknowledge
5	Revert	None	
6	Race condition	None	
7	Insufficient Gas Griefing	Medium	Acknowledge
8	The major impact of flash loans	None	
9	Unreasonable economic model	None	
10	Predictable random numbers	None	
11	Voting rights management confusion	None	

ID	Audit project	Risk level	Status
12	Privacy leak	None	
13	Improper use of time on chain	None	
14	Improper codes in fallback function	None	
15	Improper identification	None	
16	Inappropriate opcode	None	
17	Inappropriate assembly	None	
18	Constructor irregularities	None	
19	Return value irregularity	None	
20	Event irregularity	Minor	Resolved
21	Keywords irregularity	None	
22	Not following ERC standards	None	
23	Irregularity of condition judgment	Minor	Resolved
24	Risk of liquidity drain	None	
25	Centralization Risk	Medium	Acknowledge
26	Logic change risk	None	
27	Integer overflow	None	
28	Improper function visiblity	None	
29	Improper initialization of variables	None	
30	Improper contract calls	None	
31	Variable irregularities	None	
32	Replay	None	
33	Write to Arbitrary Storage Location	None	
34	Honeypot logic	None	
35	Has collision	None	
36	Improper logic in receiving awards	Medium	Acknowledge
37	Use the not recommended method	Minor	Acknowledge
38	Basic coding principles were not followed	Minor	Acknowledge

5. Risk and Modification Program

The following section provides detailed information about the risk items learned after the audit, including the type of risk, risk level, location of the issue, description of the problem, recommendations for changes, and feedback from the project owner.

Risk type	**MEV Possibility**	**Risk level**	**Major**
Location	Line 207、Line 36	**Contract file**	Treasury.sol
Description	Line 207: MEV bots can listen to the CLEAN call and run for fund transfer. Line 236: Since the index is self-increasing for array push when voting, the attacker can layout the index in advance according to the maxCircle range and listen to the count call to rob-run the transfer to repeat the vote count		
Recommedation	No main net coin voting, airdrop contract tokens with governance based on snapshot; or no self-increasing index		
Update	Project requirements must use the main network coins, and the design count will not be too high; if the attack behavior affects the voting results, it will not be implemented.		

Risk type	**Insufficient Gas Griefing**	**Risk level**	**Medium**
Location	Line 204	**Contract file**	**Treasury.sol**
Description	The existence of a possible great cycle		
Recommedation	Limit cycle times		
Update	The project owner specifies parameters. There is no real risk		

Risk type	**Irregularity of condition judgment**	**Risk level**	**Minor**
Location	All	**Contract file**	All
Description	No non-zero judgment for incoming addresses		
Recommendation	No non-zero judgment on incoming addresses to prevent contract failure due to incorrect address settings		
Update	Acknowledged and resolved		

Risk type	Event irregularity	Risk level	Minor
Location	Line 104-156	Contract file	Governance.sol
Description	Fundamental status changes are not logged		
Recommedation	Use event		
Update	Acknowledged and resolved		

Risk type	Centralization Risk	Risk level	Medium
Location	All	Contract file	Governance.sol
Description	Contract functions involving permission control, are not timelock mechanism, or multi-signature mechanism		
Recommendation	Decentralize individual private key permissions and use timelock and multi-signature mechanisms		
Update	The project plans to use the multi-signature and timelock mechanism, but it is not currently reflected in the code		

Risk type	Improper logic in receiving awards	Risk level	Medium
Location	Line 133	Contract file	Treasury.sol
Description	Reward issuance should use the mechanism of decentralized claim and should not be cyclic in the contract and the caller should pay the gas fee		
Recommendation	Use decentralized claim claiming logic		
Update	According product design requirements need to be retained, only against the code normality issues		

Risk type	Use the not recommended method	Risk level	Minor
Location	Line 378	Contract file	Treasury.sol
Description	Due to the implementation of the Istanbul upgrade, the transfer and send methods are no longer recommended		
Recommendation	Use the call method and add reentrancy protection		
Update	The project plans are deployed using proxy contracts; risk can be controlled		

Risk type	**Basic coding principles were not followed**	**Risk level**	**Minor**
Location	Line 133	**Contract file**	Treasury.sol
Description	Calling bonus issuance in the proposed method is not in line with the principle of one-thing coding		
Recommendation	Use other methods such as bookkeeping or expired zeroing		
Update	According product design requirements need to be retained, only against the code normality issues		

6. Recommendation

N/A

Disclaimer

i. This audit report focuses only on the types of audits identified in the final report issued. Other unknown security vulnerabilities are not part of this audit, and we do not accept responsibility for them.

ii. We shall only issue an audit report based on an attack or vulnerability that existed or occurred before the issuance of the audit report. We cannot determine the likely impact on the security posture of our projects for new attacks or vulnerabilities that may exist or occur in the future, and we are not responsible for them.

iii. The security audit analysis and other elements of our published audit report shall be based solely on documents and materials (including, but not limited to, contract codes) provided to us by the Project Party before the release of the audit report. Such documents and materials shall not be untrue, inaccurate, uninformative, altered, deleted, or concealed, and if the documents and materials provided by the Project Party are false, inaccurate, uninformative, changed, deleted or hidden, or if the documents and materials provided by the Project Party are untrue, inaccurate, uninformative, altered, deleted or concealed, or if the documents and materials provided by the Project Party are uninformative, uninformative, altered, deleted or hidden. If the records and information provided by the Project Party are untrue, inaccurate, uninformative, altered, deleted, or concealed, or if changes to such documents and information are made after the issuance of the audit report, we shall not be liable for any loss or adverse effect arising from any inconsistency between the reflected and actual conditions.

iv. The Project Parties are aware that our audit report is based on documents and information provided by the Project Parties and relies on the technology currently available. However, due to the technical limitations of any organization, there is a possibility that our audit report may not fully detect all risks. Our audit team encourages the project development team and any interested parties to conduct subsequent testing and audits of the project.

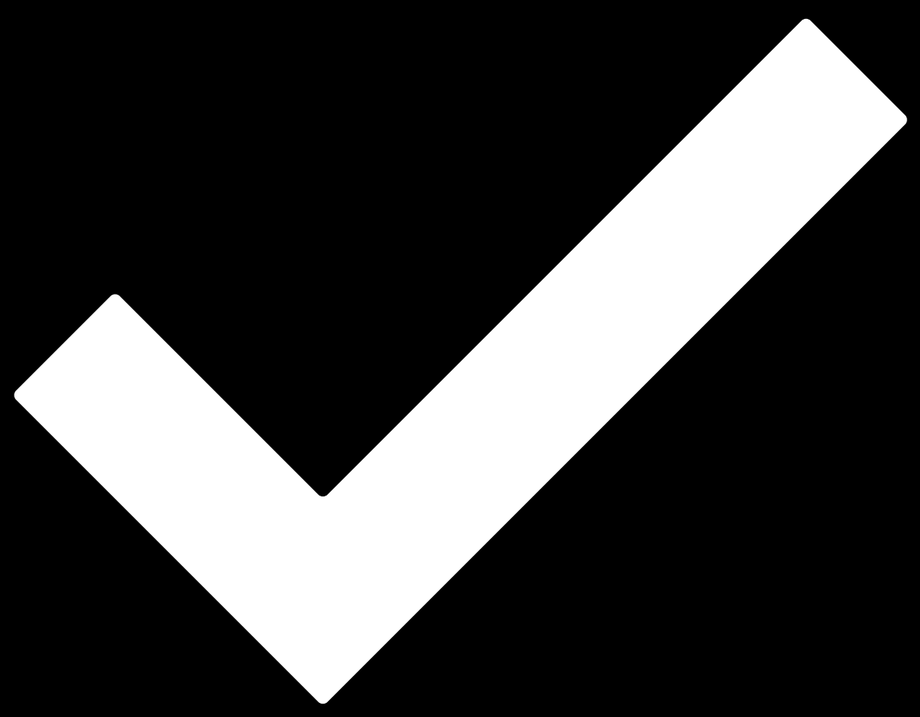
v. The project owner warrants that the project for which we are engaged to provide audit or testing services is legal, compliant, and does not violate applicable laws. The audit report is for the project owner's reference only, and the contents, manner of obtaining, use of, and any services or resources involved in the audit report shall not be relied upon for investment, tax, legal, regulatory, or advisory purposes of any kind, and we shall not be liable therefor. The Project Party shall not refer to, quote, display, or send the Audit Report in whole or in part to any third party without our prior written consent. The Project Party shall bear any loss or liability arising from that place. We assume no responsibility for any reliance on or use of the audit report for any purpose.

vi. This audit report does not cover the compiler of the contract or any areas beyond the programming language of the Smart Contract. The risk and liability of the audited Smart Contract arising from references to off-chain information or resources is the sole responsibility of the project party.

vii. Force Majeure. Force majeure means an unforeseen event whose occurrence and consequences cannot be avoided and cannot be overcome by the parties at the time of entering into the contract, including but not limited to natural disasters such as war, typhoon, flood, fire, earthquake, tidal wave, lightning, natural disaster, strike, nuclear explosion, epidemic and other unforeseen events such as changes in laws, regulations and policies and governmental acts, whose occurrence and consequences cannot be prevented or avoided, and which contains, affects or delays the performance by either party of all or part of its obligations under the contract.

viii. Suppose either party believes that the occurrence of force majeure affects the performance of its obligations under this Agreement. In that case, it shall promptly notify the other party and, depending on the extent of the effect of the event on the performance of the Agreement; the parties shall consult to determine whether to terminate the Agreement or partially relieve itself of its obligations to perform the Agreement, or to extend the performance of the Agreement.

ix. In force majeure, neither party shall be deemed in breach or non-performance of its obligations under this Agreement. Any financial commitments existing before the event shall not be affected, and the project party shall make payment for work performed by us.



PASSED.

DATE 29 Mar 2022

AUDITOR 歐科雲鏈

This audit aimed to review the Treasury proposal voting governance and award issuance functionality written in Solidity language for the OKC Treasury project, examine its design architecture, identify potential security risks, and attempt to find possible vulnerabilities.