Cyberscope

Audit Report
BlaroThings

April 2025

SHA256 38a906883726da849d25fc8760cf03fdf4d1bf5a96e74e55b6ab51bbddacb74c

Audited by © cyberscope

Table of Contents

Risk Classification

The criticality of findings in Cyberscope's smart contract audits is determined by evaluating multiple variables. The two primary variables are:

1. **Likelihood of Exploitation**: This considers how easily an attack can be executed, including the economic feasibility for an attacker.
2. **Impact of Exploitation**: This assesses the potential consequences of an attack, particularly in terms of the loss of funds or disruption to the contract's functionality.

Based on these variables, findings are categorized into the following severity levels:

1. **Critical**: Indicates a vulnerability that is both highly likely to be exploited and can result in significant fund loss or severe disruption. Immediate action is required to address these issues.
2. **Medium**: Refers to vulnerabilities that are either less likely to be exploited or would have a moderate impact if exploited. These issues should be addressed in due course to ensure overall contract security.
3. **Minor**: Involves vulnerabilities that are unlikely to be exploited and would have a minor impact. These findings should still be considered for resolution to maintain best practices in security.
4. **Informative**: Points out potential improvements or informational notes that do not pose an immediate risk. Addressing these can enhance the overall quality and robustness of the contract.

Severity	Likelihood / Impact of Exploitation
● Critical	Highly Likely / High Impact
● Medium	Less Likely / High Impact or Highly Likely/ Lower Impact
● Minor / Informative	Unlikely / Low to no Impact

Review

Audit Updates

Initial Audit	13 Jan 2025
Corrected Phase 2	23 Jan 2025
Corrected Phase 3	08 Apr 2025
Corrected Phase 4	22 Apr 2025

Source Files

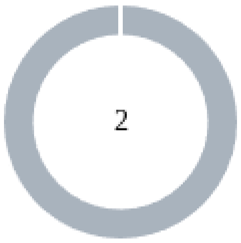
Filename	SHA256
DAO.sol	38a906883726da849d25fc8760cf03fdf4d1bf5a96e74e55b6ab51bbddacb74c

Overview

The DAOGovernance contract is a decentralized governance system designed to manage proposals and voting within a DAO (Decentralized Autonomous Organization). The contract allows the owner to create proposals that include details such as a description, an address for potential fund transfers, an Ether transfer amount, and an optional state-change action. Proposals can then be voted on by participants with assigned voting power. Each voter can contribute their voting weight, and proposals are executed if they receive a minimum number of votes (`MINIMUM_VOTES_REQUIRED`), ensuring a degree of consensus before implementation.

Key features include the ability to transfer Ether to a specified recipient if a proposal dictates such an action. The contract ensures security through mechanisms like OpenZeppelin's ReentrancyGuard to prevent reentrancy attacks and Ownable to restrict sensitive operations to the owner. Voting power is dynamically assigned by the owner, and each participant's voting history is tracked to prevent duplicate votes on the same proposal. Additionally, the contract supports Ether deposits and uses SafeMath for safe arithmetic operations, enhancing its reliability.

Findings Breakdown



●	Critical	0
●	Medium	0
●	Minor / Informative	2

Severity		Unresolved	Acknowledged	Resolved	Other
●	Critical	0	0	0	0
●	Medium	0	0	0	0
●	Minor / Informative	2	0	0	0

Diagnostics

🔴 Critical 🟠 Medium ● Minor / Informative

Severity	Code	Description	Status
●	CCR	Contract Centralization Risk	Unresolved
●	PTAI	Potential Transfer Amount Inconsistency	Unresolved

CCR - Contract Centralization Risk

Criticality	Minor / Informative
Location	DAO.sol#L100,134,144
Status	Unresolved

Description

The contract's functionality and behavior are heavily dependent on external parameters or configurations. While external configuration can offer flexibility, it also poses several centralization risks that warrant attention. Centralization risks arising from the dependence on external configuration include Single Point of Control, Vulnerability to Attacks, Operational Delays, Trust Dependencies, and Decentralization Erosion.

The contract allows any user to buy voting power until a certain threshold. However, this threshold could exceed the minimum votes required for executing a proposal, enabling them to unilaterally create a proposal, vote on it, and execute it with minimal or no participation from other voters. This behavior undermines the decentralized governance principles of a DAO, as decisions can be manipulated by a malicious user without requiring consensus from the broader community.

```
function createProposal(
    string memory description,
    address recipient,
    uint256 amount
) external onlyVoters {
    require(bytes(description).length > 0, "Proposal description cannot be empty");
    require(votingPower[msg.sender].amount >= MINIMUM_VOTES_REQUIRED, "Insufficient
voting power");
    ...
}
...
proposal.voteCount += votingPower[msg.sender].amount;
...
function executeProposal(uint256 proposalId) external nonReentrant {
    require(proposalId <= proposalCount, "Invalid proposalId");
    Proposal storage proposal = proposals[proposalId];
    require(!proposal.executed, "Proposal already executed");
    require(proposal.voteCount >= MINIMUM_VOTES_REQUIRED, "Not enough votes");
    ...
}
```

Recommendation

To address this finding and mitigate centralization risks, it is recommended to evaluate the feasibility of migrating critical configurations and functionality into the contract's codebase itself. This approach would reduce external dependencies and enhance the contract's self-sufficiency. It is essential to carefully weigh the trade-offs between external configuration flexibility and the risks associated with centralization.

PTAI - Potential Transfer Amount Inconsistency

Criticality	Minor / Informative
Location	DAO.sol#L68,72,76
Status	Unresolved

Description

The `transfer()` and `transferFrom()` functions are used to transfer a specified amount of tokens to an address. The fee or tax is an amount that is charged to the sender of an ERC20 token when tokens are transferred to another address. According to the specification, the transferred amount could potentially be less than the expected amount. This may produce inconsistency between the expected and the actual behavior.

The following example depicts the diversion between the expected and actual amount.

Tax	Amount	Expected	Actual
No Tax	100	100	100
10% Tax	100	100	90

```
require(governanceToken.transferFrom(msg.sender, address(this), amount), "Token
transfer failed");

uint256 lockEndTime = block.timestamp + VOTING_POWER_LOCK_PERIOD;
votingPower[msg.sender] = VotingPower({
    amount: currentPower + amount,
    lockEndTime: lockEndTime
});

totalVotingPower += amount;
```

Recommendation

The team is advised to take into consideration the actual amount that has been transferred instead of the expected.

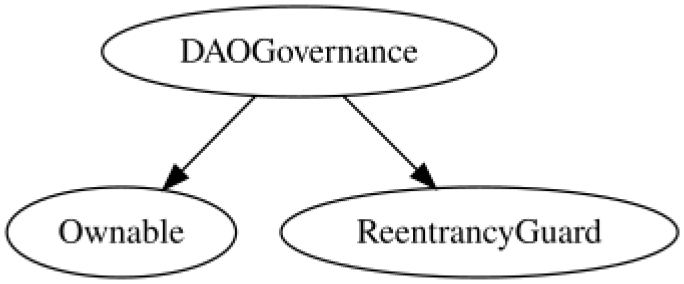
It is important to note that an ERC20 transfer tax is not a standard feature of the ERC20 specification, and it is not universally implemented by all ERC20 contracts. Therefore, the contract could produce the actual amount by calculating the difference between the transfer call.

```
Actual Transferred Amount = Balance After Transfer - Balance Before Transfer
```

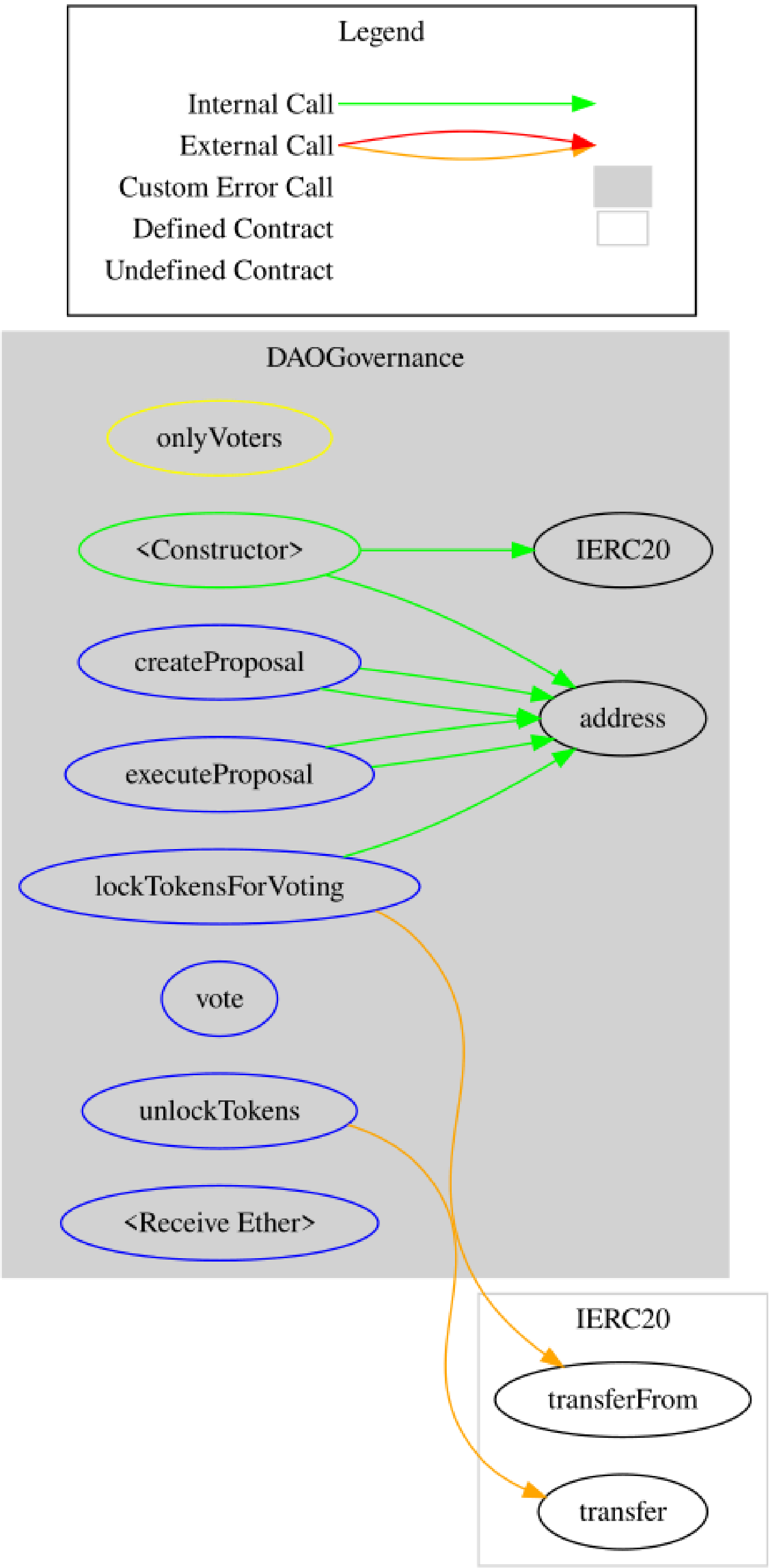
Functions Analysis

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
DAOGovernance	Implementation	Ownable, ReentrancyGuard		
		Public	✓	Ownable
	lockTokensForVoting	External	✓	nonReentrant
	unlockTokens	External	✓	nonReentrant
	createProposal	External	✓	onlyVoters
	vote	External	✓	onlyVoters
	executeProposal	External	✓	nonReentrant
		External	Payable	-

Inheritance Graph



Flow Graph


Legend
Internal Call
External Call
Custom Error Call
Defined Contract
Undefined Contract
DAOGovernance
onlyVoters
<Constructor>
IERC20
createProposal
address
executeProposal
lockTokensForVoting
vote
unlockTokens
<Receive Ether>
IERC20
transferFrom
transfer

Summary

BlaroThings contract implements a governance mechanism. This audit investigates security issues, business logic concerns, and potential improvements.

Disclaimer

The information provided in this report does not constitute investment, financial or trading advice and you should not treat any of the document's content as such. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes nor may copies be delivered to any other person other than the Company without Cyberscope's prior written consent. This report is not nor should be considered an "endorsement" or "disapproval" of any particular project or team. This report is not nor should be regarded as an indication of the economics or value of any "product" or "asset" created by any team or project that contracts Cyberscope to perform a security assessment. This document does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors' business, business model or legal compliance. This report should not be used in any way to make decisions around investment or involvement with any particular project. This report represents an extensive assessment process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk Cyberscope's position is that each company and individual are responsible for their own due diligence and continuous security Cyberscope's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies and in no way claims any guarantee of security or functionality of the technology we agree to analyze. The assessment services provided by Cyberscope are subject to dependencies and are under continuing development. You agree that your access and/or use including but not limited to any services reports and materials will be at your sole risk on an as-is where-is and as-available basis Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives false negatives and other unpredictable results. The services may access and depend upon multiple layers of third parties.

About Cyberscope

Cyberscope is a blockchain cybersecurity company that was founded with the vision to make web3.0 a safer place for investors and developers. Since its launch, it has worked with thousands of projects and is estimated to have secured tens of millions of investors' funds.

Cyberscope is one of the leading smart contract audit firms in the crypto space and has built a high-profile network of clients and partners.

The Cyberscope team

cyberscope.io