Cyberscope

Audit Report

Policy

July 2022

SHA256 5476a71576bbd6b3f1db3835f8e1c4594cab88d4275e7818a67a4d1a9a1631ec

Audited by © cyberscope

Table of Contents

Table of Contents 1
Contract Review 3
Audit Updates 3
Source Files 4
Introduction 6
Contract Diagnostics 7
CR - Code Repetition 8
Description 8
Recommendation 8
DPI - Decimals Potential Inconsistency 9
Description 9
Recommendation 9
L01 - Public Function could be Declared External 10
Description 10
Recommendation 10
L02 - State Variables could be Declared Constant 11
Description 11
Recommendation 11
L04 - Conformance to Solidity Naming Conventions 12
Description 12
Recommendation 12
L11 - Unnecessary Boolean equality 13
Description 13
Recommendation 13
Contract Functions 14
Contract Flow 17

Domain Info 18

Summary 19

Disclaimer 20

About Cyberscope 21

Contract Review

Contract Name	Policy
Explorer	https://bscscan.com/token/0x0551EBe151A0AB86911ff1986cA16e22d65c0Cef
Domain	https://defilabs.farm

Audit Updates

Initial Audit	19th July 2022
Corrected	

Source Files

Filename	SHA256
@openzeppelin/contracts/access/Ownable.sol	4249cf7ca3111e505f92c0f23c5afcf3ec757055b0bc2babb7a637563ec7ddfd
@openzeppelin/contracts/GSN/Context.sol	0d30be46514c535ec373b07a5041ac484ca66a18fc66feb2b68c0e21103f1c01
@openzeppelin/contracts/math/Math.sol	188b811abb02569bcef02b41119b7653008c97d0609bbb1151e7e18c0ccc310e
@openzeppelin/contracts/math/SafeMath.sol	665f1eab7288dc1142b1330d74a42cf18bb24d1d9fbf1efbb17e0acb46a278dd
@openzeppelin/contracts/token/ERC20/IERC20.sol	d63052248b744c9a434cfb6feb4ac10aa5e4b9b852f728439777240b6af46b6d
@openzeppelin/contracts/token/ERC20/SafeERC20.sol	c4068e540290b83c45a8e52f2747de6b6ada924696f2499454df35633a4d4171
@openzeppelin/contracts/utils/Address.sol	23abcffc4cb1dc1d471500a17ba601ac92319f2d152ace49add276819d78d8cd
@openzeppelin/contracts/utils/ReentrancyGuard.sol	0bfdff81c9989c48ac53b71f89802cc37ee4774acaa6a6a33cab06195774fb26
contracts/interfaces/IEIP20.sol	8239f179f6b0e97e0588964cfe7b2ab2b8c7233caa692a5188fec144a2ff63d9
contracts/interfaces/IOracle.sol	876aff9492ecdae57325277ba4319b3f4364bf21b69cb19bbc0ecaaee3052f92

contracts/Policy.sol	5476a71576bbd6b3f1db3835f8e1c4594cab88d4275e7818a67a4d1a9a1631ec

Introduction

Policy main functionality is to initiate the vPool5 pool. It adds 6 pools with rewards in:

- CAKE
- BNB
- USDT
- BUSD
- BTC
- ETH

Each pool is initialized with 4 different lock duration options:

- 1 day
- 7 days
- 30 days
- 60 days

Contract Diagnostics

Critical Medium Minor

Severity	Code	Description
	CR	Code Repetition
	DCI	Decimals Potential Inconsistency
	L01	Public Function could be Declared External
	L02	State Variables could be Declared Constant
	L04	Conformance to Solidity Naming Conventions
	L11	Unnecessary Boolean equality

CR - Code Repetition

Criticality	minor
Location	contract.sol#L41,L49,L56,L63,L70

Description

There are code segments that are repetitive in the contract. Those segments increase the code size of the contract unnecessarily.

```
        //cake
        IPool(pool).addPool(cake, cake_min * 10**usdtDecimals, cake_maxbenefit,
PolicyType.BenefitType.T7);
        uint256 cake_pid = IPool(pool).poolIncr();
        IPool(pool).setPool(cake_pid, PolicyType.StakeType.Day1,  cake_day1);
        IPool(pool).setPool(cake_pid, PolicyType.StakeType.Day7,  cake_day7);
        IPool(pool).setPool(cake_pid, PolicyType.StakeType.Day30, cake_day30);
        IPool(pool).setPool(cake_pid, PolicyType.StakeType.Day60, cake_day60);

        //bnb
        IPool(pool).addPool(bnb, bnb_min * 10**usdtDecimals, bnb_maxbenefit,
PolicyType.BenefitType.T7);
        uint256 bnb_pid = IPool(pool).poolIncr();
        IPool(pool).setPool(bnb_pid, PolicyType.StakeType.Day1,  bnb_day1);
        IPool(pool).setPool(bnb_pid, PolicyType.StakeType.Day7,  bnb_day7);
        IPool(pool).setPool(bnb_pid, PolicyType.StakeType.Day30, bnb_day30);
        IPool(pool).setPool(bnb_pid, PolicyType.StakeType.Day60, bnb_day60);
```

Recommendation

Create an internal function that contains the code segment and remove it from all the sections.

DPI - Decimals Potential Inconsistency

Criticality	minor
Location	contract.sol#L93

Description

The contract initialize the variable usdtDecimals with a plain number. The _usdt is variable and not a fixed value. As a result it may produce inconsistency between the expected and the actual decimals.

```
usdtDecimals = 18;
```

Recommendation

The usdtDecimals should be initialized by the origin decimals source.
A potential implementation could be:

```
usdtDecimals = ERC20(_usdt).decimals()
```

L01 - Public Function could be Declared External

Criticality	minor
Location	contracts/Policy.sol#L103

Description

Public functions that are never called by the contract should be declared external to save gas.

```
setPolicys
```

Recommendation

Use the external attribute for functions never called from the contract.

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contracts/Policy.sol#L57,49,64,78,42,43,70,63,77,56,50,71

Description

Constant state variables should be declared constant to save gas.

```
btc_maxbenefit
bnb_maxbenefit
usdt_min
eth_min
busd_min
btc_min
cake_maxbenefit
cake_min
eth_maxbenefit
...
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contracts/Policy.sol#L57,78,50,60,65,74,68,82,73,51,46,79,53,70,72,42,47,67,66,45,54,63,61,64,58,75,80,71,49,77,43,52,81,56,44,59

Description

Solidity defines a naming convention that should be followed. Rule exceptions:

- Allow constant variable name/symbol/decimals to be lowercase.
- Allow _ at the beginning of the mixed_case match for private variables and unused parameters.

```
usdt_day7
cake_day1
usdt_min
eth_day30
bnb_day7
cake_maxbenefit
eth_min
bnb_min
btc_maxbenefit
...
```

Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions.

L11 - Unnecessary Boolean equality

Criticality	minor
Location	contracts/Policy.sol#L103

Description

The comparison to boolean constants is redundant. Boolean constants can be used directly and do not need to be compared to true or false.

```
require(bool,string)(isDone == false,Policy: is done)
```

Recommendation

Remove the equality to the boolean constant.

Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
Ownable	Implementation	Context		
	<Constructor>	Internal	✓	
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
Math	Library			
	max	Internal		
	min	Internal		
	average	Internal		
SafeMath	Library			
	add	Internal		
	sub	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	div	Internal		
	mod	Internal		
	mod	Internal		
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-

	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
SafeERC20	Library			
	safeTransfer	Internal	✓	
	safeTransferFrom	Internal	✓	
	safeApprove	Internal	✓	
	safeIncreaseAllowance	Internal	✓	
	safeDecreaseAllowance	Internal	✓	
	_callOptionalReturn	Private	✓	
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	_functionCallWithValue	Private	✓	
ReentrancyGuard	Implementation			
	<Constructor>	Internal	✓	
IEIP20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	name	External		-
	symbol	External		-
	decimals	External		-
	allowance	External		-
	approve	External	✓	-
IOracle	Interface			

	R	External		-
PolicyType	Library			
IPool	Interface			
	poolIncr	External		-
	addPool	External	✓	-
	setPool	External	✓	-
Policy	Implementation	Ownable, Reentrancy Guard		
	<Constructor>	Public	✓	-
	setPolicys	Public	✓	onlyOwner

Contract Flow

Domain Info

Domain Name	defilabs.farm
Registry Domain ID	d44f7165186c43e6ab7e5570545b2f9e-DONUTS
Creation Date	2021-09-23T12:54:45Z
Updated Date	2022-07-18T09:44:52Z
Registry Expiry Date	2024-09-23T12:54:45Z
Registrar WHOIS Server	http://whois.cloudflare.com
Registrar URL	http://cloudflare.com
Registrar	Cloudflare, Inc
Registrar IANA ID	1910

The domain has been created in about 2 years before the creation of the audit.

There is no public billing information, the creator is protected by the privacy settings.

Summary

Policy responsible for the initiation of the vPool5 contract. In other words policy is the operator for the vPool5 contract. The Smart Contract analysis reported no compiler error or critical issues.

Disclaimer

All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment.

Cyberscope team provides no guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed.

The Cyberscope team has audited this project for general information and only expresses their opinion based on similar projects and checks from popular diagnostic tools. Under no circumstances did Cyberscope receive a payment to manipulate those results or change the awarding badge that we will be adding in our website.

Always Do your own research and protect yourselves from scams. This document should not be presented as a reason to buy or not buy any particular token.

The Cyberscope team disclaims any liability for the resulting losses.

About Cyberscope

Coinscope audit and K.Y.C. service has been rebranded to Cyberscope.

Coinscope is the leading early coin listing, voting and auditing authority firm. The audit process is analyzing and monitoring many aspects of the project. That way, it gives the community a good sense of security using an informative report and a generic score.

Cyberscope and Coinscope are aiming to make crypto discoverable and efficient globally. They provide all the essential tools to assist users draw their own conclusions.

The Cyberscope team

https://www.cyberscope.io