

# Audit Report **Seed**

October 2022

Github https://github.com/moonappxxx/moonapp-contracts

Commit 849ced7b7b96948529d46f9295717ec5a1620f4f

Audited by © cyberscope



# **Table of Contents**

Table of Contents	1
Introduction	2
Roles	2
Contract Review	3
Audit Updates	3
Source Files	4
Contract Diagnostics	6
BLC - Business Logic Concern	7
Description	7
Recommendation	7
MC - Missing Check	8
Description	8
Recommendation	9
L04 - Conformance to Solidity Naming Conventions	10
Description	10
Recommendation	10
Contract Functions	11
Contract Flow	16
Domain Info	17
Summary	18
Disclaimer	19
About Cyberscope	20



## Introduction

The Seed contract manages the vesting contracts for the investors.

#### Roles

The "admin" role has the ability to vest a number of tokens for each investor. The investors can be added until the initialization of the vesting.



# **Contract Review**

Contract Name	Seed
Compiler Version	v0.8.11+commit.d7f03943
Github	https://github.com/moonappxxx/moonapp-contracts
Commit	849ced7b7b96948529d46f9295717ec5a1620f4f
Testing Deploy	https://testnet.bscscan.com/token/0xD1EB5926261f849 d633e7DdBBAa25DD40AB7Bf49
Domain	https://moonapp.org

# **Audit Updates**

Initial Audit	3rd October 2022 <a href="https://github.com/cyberscope-io/audits/blob/main/1-xxx/v1/seed.pdf">https://github.com/cyberscope-io/audits/blob/main/1-xxx/v1/seed.pdf</a>
Corrected	7th October 2022



# Source Files

Filename	SHA256
@openzeppelin/c ontracts/access/ Ownable.sol	9353af89436556f7ba8abb3f37a6677249aa4df6024fbfaa9 4f79ab2f44f3231
@openzeppelin/c ontracts/token/E RC20/ERC20.sol	5031430cc2613c32736d598037d3075985a2a09e61592a 013dbd09a5bc2041b8
@openzeppelin/c ontracts/token/E RC20/extensions /draft-IERC20Per mit.sol	3e7aa0e0f69eec8f097ad664d525e7b3f0a3fda8dcdd97de 5433ddb131db86ef
@openzeppelin/c ontracts/token/E RC20/extensions /ERC20Burnable. sol	0344809a1044e11ece2401b4f7288f414ea41fa9d1dad24 143c84b737c9fc02e
@openzeppelin/c ontracts/token/E RC20/extensions /IERC20Metadat a.sol	af5c8a77965cc82c33b7ff844deb9826166689e55dc037a 7f2f790d057811990
@openzeppelin/c ontracts/token/E RC20/IERC20.sol	94f23e4af51a18c2269b355b8c7cf4db8003d075c9c5410 19eb8dcf4122864d5
@openzeppelin/c ontracts/token/E RC20/utils/SafeE RC20.sol	fa36a21bd954262006d806b988e4495562e7b50420775e 2aa0deecb596fd1902
@openzeppelin/c ontracts/utils/Ad dress.sol	1e0922f6c0bf6b1b8b4d480dcabb691b1359195a297bde 6dc5172e79f3a1f826



@openzeppelin/c ontracts/utils/Co ntext.sol	1458c260d010a08e4c20a4a517882259a23a4baa0b5bd9 add9fb6d6a1549814a
@openzeppelin/c ontracts/utils/ma th/Math.sol	929523c09910460ad708c75878d89b9fbed12b65cb5d8b 670200c793131072f4
@openzeppelin/c ontracts/utils/ma th/SafeMath.sol	0dc33698a1661b22981abad8e5c6f5ebca0dfe5ec149163 69a2935d888ff257a
contracts/Govern ed.sol	216f03644d4e517caba4b44b8f3b74c358462601918a7be 264790ef1cc1bde4c
contracts/Moona ppToken.sol	bc938fcfe911c7df6871209c77bf053a2f1b4b9bab19f7c6 d45a4feeae03e9d6
contracts/Seed.s	fe4c012fc5babafc0ba3946fccc77d3b6b25b7f05cef69791 a56f508be63fd27
contracts/Token Vesting.sol	66f193371f6346b5fa85acf98fbadcd29c4b5e7468728ab9 4d4f341d1e3c0fdc



# **Contract Diagnostics**

CriticalMediumMinor / Informative

Severity	Code	Description	Status
•	BLC	Business Logic Concern	Unresolved
•	MC	Missing Check	Unresolved
•	L04	Conformance to Solidity Naming Conventions	Unresolved



### BLC - Business Logic Concern

Criticality	minor / informative
Location	contract.sol#L82
Status	Unresolved

#### Description

The business logic seems peculiar. The implementation may not follow the expected behavior.

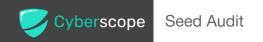
The token should give permission (approve) over the Seed address before transfer transactions.

```
SafeERC20.safeTransfer(
    IERC20(token),
    address(vesting),
    tokensAmount
);
```

#### Recommendation

The team is advised to carefully check if the implementation follows the expected business logic.

The contract could approve the entire vesting amount before transfer transcactions.



#### MC - Missing Check

Criticality	minor / informative
Location	contract.sol#L47,57,27
Status	Unresolved

#### Description

The contract is processing variables that have not properly sanitized and checked that they form the proper shape. These variables may produce vulnerability issues.

The contract is not taking into consideration all the invested amount.

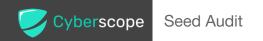
```
require(

amount <= availableTokens,

"ADD_INVESTOR: not enought tokens left."
);
```

The following variables \_cliff, \_releaseRate, \_initialReleaseRate are not properly sanitized before the vesting token initialization.

```
function releaseTokens(
    uint256 _start,uint256 _cliff, uint256 _releaseRate, uint256 _initialReleaseRate
) external {
    //..
    uint256 initialReleaseAmont = (tokensAmount / 100) *
        _initialReleaseRate; // release % of the tokens on listing
    TokenVesting vesting = new TokenVesting(
        investors[i],
        startTime,
        _cliff,
        _releaseRate,
        initialReleaseAmont
);
```



The variable availableTokens is not properly sanitized before its initialization in the contractor.

```
constructor(address tokenAddress, uint256 _availableTokens) {
  token = MoonappToken(tokenAddress);
  admin = msg.sender;

  availableTokens = _availableTokens * (10**18);
}
```

#### Recommendation

The contract should properly check the variables according to the required specifications.

The contract should check if the total amount of tokens across investors is less than the available tokens.

It is recommended to pre-check variables before using them with other contracts.

- The variable start has to be greater than the current timestamp.
- The variable cliff has to be greater than zero
- The variable releaseRate have to be lower than 100%

The contract could embody a check on the contractor for the availableTokens variable to be greater than zero in order to avoid initializing the contract with zero available amount.



# L04 - Conformance to Solidity Naming Conventions

Criticality	minor / informative
Location	contracts/Seed.sol#L96,104
Status	Unresolved

#### Description

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

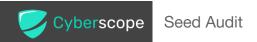
- Allow constant variable name/symbol/decimals to be lowercase.
- Allow \_ at the beginning of the mixed\_case match for private variables and unused parameters.

\_investor

#### Recommendation

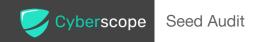
Follow the Solidity naming convention.

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



# **Contract Functions**

Contract	Туре	Bases		
	Function Name	Visibility	Mutability	Modifiers
Own abla	langle or and others	Contact		
Ownable	Implementation	Context		
	<constructor></constructor>	Public	<b>√</b>	-
	owner	Public		-
	_checkOwner	Internal		
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	<b>√</b>	onlyOwner
	_transferOwnership	Internal	<b>√</b>	
ERC20	Implementation	Context, IERC20, IERC20Met adata		
	<constructor></constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	1	-
	transferFrom	Public	<b>✓</b>	-
	increaseAllowance	Public	1	-
	decreaseAllowance	Public	1	-
	_transfer	Internal	1	
	_mint	Internal	1	
	_burn	Internal	✓	
	_approve	Internal	✓	
	_spendAllowance	Internal	/	



	_beforeTokenTransfer	Internal	✓	
	_afterTokenTransfer	Internal	1	
IERC20Permit	Interface			
	permit	External	1	-
	nonces	External		-
	DOMAIN_SEPARATOR	External		-
ERC20Burnabl e	Implementation	Context, ERC20		
	burn	Public	✓	-
	burnFrom	Public	✓	-
IERC20Metad ata	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	External		-
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	1	-
	allowance	External		-
	approve	External	1	-
	transferFrom	External	1	-
SafeERC20	Library			
	safeTransfer	Internal	1	
	safeTransferFrom	Internal	✓	
	safeApprove	Internal	✓	
	safeIncreaseAllowance	Internal	✓	
	safeDecreaseAllowance	Internal	1	
	safePermit	Internal	✓	
	_callOptionalReturn	Private	1	

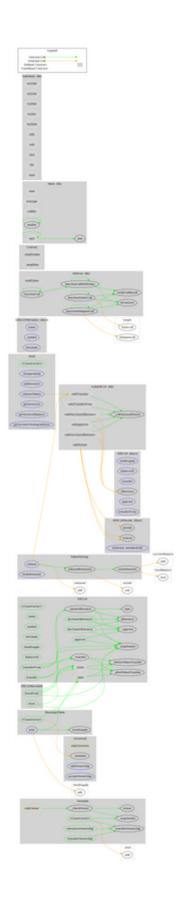
13

Address	Library			
	isContract	Internal		
	sendValue	Internal	1	
	functionCall	Internal	1	
	functionCall	Internal	1	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	1	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	1	
	functionDelegateCall	Internal	1	
	verifyCallResult	Internal		
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
Math	Library			
	max	Internal		
	min	Internal		
	average	Internal		
	ceilDiv	Internal		
	mulDiv	Internal		
	mulDiv	Internal		
	sqrt	Internal		
	sqrt	Internal		
SafeMath	Library			
	tryAdd	Internal		
	trySub	Internal		
	tryMul	Internal		
	tryDiv	Internal		
	tryMod	Internal		
	add	Internal		



	sub	Internal		
	mul	Internal		
	div	Internal		
	mod	Internal		
	sub	Internal		
	div	Internal		
	mod	Internal		
Governed	Implementation			
	_initialize	Internal	1	
	addOwnership	External	<b>✓</b>	onlyGovernor
	acceptOwnership	External	1	-
MoonappToke n	Implementation	ERC20, ERC20Burn able, Governed		
	<constructor></constructor>	Public	1	ERC20
	burnFrom	Public	1	onlyGovernor
	mint	External	<b>√</b>	onlyGovernor
Seed	Implementation			
	<constructor></constructor>	Public	1	-
	changeAdmin	External	1	-
	addInvestor	External	1	-
	releaseTokens	External	<b>√</b>	-
	getInvestors	External		-
	getInvestorBalance	External		-
	getInvestorVestingAddress	External		-
TokenVesting	Implementation	Ownable		
	<constructor></constructor>	Public	1	-
	release	External	<b>√</b>	-
	releasableAmount	Public		-
	vestedAmount	Public		-
	lockedAmount	External		-

# Contract Flow





# Domain Info

Domain Name	moonapp.org
Registry Domain ID	ebf9cc2ae696406f89ddb496f15a1e47-LROR
Creation Date	2022-01-23T16:44:59Z
Updated Date	2022-03-25T03:49:23Z
Registry Expiry Date	2023-01-23T16:44:59Z
Registrar WHOIS Server	http://whois.reg.com
Registrar URL	http://www.reg.com
Registrar	Registrar of Domain Names REG.RU LLC
Registrar IANA ID	1606

The domain was created 8 months before the creation of the audit. It will expire in 4 months.

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



# Summary

The smart contract analysis reported no critical or compiler issues. This audit focused on investigating security issues and potential improvements.



#### Disclaimer

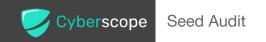
All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment.

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.

