Cyberscope

# Audit Report

# Ring Risk Free Note

April 2022

| | |
|---|---|
| Type | BEP20 |
| Network | BSC |
| Address | 0x719d9834546a5D82f80A4879C43F8620564C9E72 |
| | 0x40F9A62842A5677f3B741502cAffB1d0546b866C |
| | 0x092094687b8EDFC8ac776Dce305590b9e8Bc09C7 |
| | 0xacA4730C5ed5B11E2d134296DffD2e9BF3c27ff4 |
| Audited by | © cyberscope |

# Table of Contents

# Contract Review

## Contract 1

| | |
|---|---|
| **Contract Name** | RiskFreeNote |
| **Compiler Version** | v0.7.5+commit.eb77ed08 |
| **Optimization** | 200 runs |
| **Licence** | GNU AGPLv3 |
| **Explorer** | https://bscscan.com/address/0x719d9834546a5D82f80A4879C43F8620564C9E72 |
| **Symbol** | RFN15 |
| **BUSD interest** | 0.6% |
| **WRING interest** | 0.06% |
| **Lock Period** | 15 Days |

# Contract 2

| | |
|---|---|
| **Contract Name** | RiskFreeNote |
| **Compiler Version** | v0.7.5+commit.eb77ed08 |
| **Optimization** | 200 runs |
| **Licence** | GNU AGPLv3 |
| **Explorer** | https://bscscan.com/address/0x40F9A62842A5677f3B741502cAffB1d0546b866C |
| **Symbol** | RFN30 |
| **BUSD interest** | 2.40% |
| **WRING interest** | 0.24% |
| **Lock Period** | 30 Days |

# Contract 3

| | |
|---|---|
| **Contract Name** | RiskFreeNote |
| **Compiler Version** | v0.7.5+commit.eb77ed08 |
| **Optimization** | 200 runs |
| **Licence** | GNU AGPLv3 |
| **Explorer** | https://bscscan.com/address/0x092094687b8EDFC8ac776Dce305590b9e8Bc09C7 |
| **Symbol** | RFN90 |
| **BUSD interest** | 21.5% |
| **WRING interest** | 2.15% |
| **Lock Period** | 90 Days |

# Contract 4

| | |
|---|---|
| **Contract Name** | RiskFreeNote |
| **Compiler Version** | v0.7.5+commit.eb77ed08 |
| **Optimization** | 200 runs |
| **Licence** | GNU AGPLv3 |
| **Explorer** | https://bscscan.com/address/0xacA4730C5ed5B11E2d134296DffD2e9BF3c27ff4 |
| **Symbol** | RFN18 |
| **BUSD interest** | 87.5% |
| **WRING interest** | 8.75% |
| **Lock Period** | 180 Days |

# Source Files

| Filename | SHA256 |
|---|---|
| Address.sol | bdeba36a04b1dcde72faef89046ab0f8c7c2995f8e951a8f3e6484df96ddab6f |
| Context.sol | b78a7540c4ef34c5f6c2df250fe754eeecc3a499b88ee6b3ff7b7359653625d7 |
| Counters.sol | ff6f071aebccc21fb4f1861df6a2ffa5f84e73c903af6f91b8876cd2cd42e573 |
| EnumerableMap.sol | 8528d0f107df132496cfe638034fe96184f2f31a0458c14d3458453c1a0e9c80 |
| EnumerableSet.sol | b9fdde4c18a1aef893725bd48d9b45d7847a6cedc0879a30bea8c5d06da37213 |
| ERC165.sol | db97d5fefd0ffb9216d288026fbe5e5d4cffc5e7cb6935e66c8df19058f1b723 |
| ERC721.sol | e30efda7577455502e4b8ef29e8197afb8d0685be22e5792d13b26e8ed91f0a4 |
| IERC165.sol | 46d50c1f92ff733d9334183d3ccc4155c78fc407c88e1c34e5cf7caa0a2d7619 |
| IERC20.sol | 46de20647cd28f0049b13a1b4dec930518ac21512fbdd194b03b1f7b038da81e |
| IERC721.sol | 228b16f7a1423b6011ac657e2418db0f15f0c9118b44ef1d97c3ee1dc417a73f |
| IERC721Enumerable.sol | 589c7b50818a280aee90621be5a53285fd849ee346433ab5d8f664ed129b9af7 |
| IERC721Metadata.sol | 461687961523fc6fe9dbeb599cd697c745f55aded6d10df2e37e80a088bd79c8 |
| IERC721Receiver.sol | c4664b9064a6473b33efc9e729d29b648d4c218b96b8ac7cba8522a798d0481b |

| | |
|---|---|
| **Ownable.sol** | 8dc3f1130119fc09f57d1dd0d29c5bdeff0c2456410de8e86bf1f78f46d20cce |
| **RingNote.sol** | cac791017990217afb9eaae73d83ec8c0dffaadb4a2790c93e480e1c4d1a9a71 |
| **SafeMath.sol** | a13b39c603672312e891044a1f97c6f78fe3bb821983b5e2cd15fd2bed8b21c8 |
| **Strings.sol** | 02657c7d4fc428aa536bbda0d83f9f7d9b0bd3eeaf31bdfd6188920aefe1f9e7 |

# Audit Updates

| | |
|---|---|
| **Initial Audit** | 6th April 2022 |
| **Corrected** | |

# Contract Analysis

- Users have the ability to mint an NFT by paying an amount in WRING.
- The amount is locked for a specific period.
- After this period, the users can burn their NFTs and receive the amount back with some additional rewards.
- The rewards are WRING tokens and busd.
- The percentage of the awarded amount and the locked period are predefined by the contract.
- There are four counteract with different configurations, read more in the Contract Review section.
- The NFT is not actually locked, it can be transferred to another wallet. That means that the last wallet that holds the NFT will receive the rewards.
- The contract guarantees that the awarded amount exists before letting the user mint an NFT. Hence

# Contract Owner Privileges

The contract owner has the ability to allow the users to unlock all the NFTs even if the unlock period has not elapsed.

# Contract Diagnostics

Critical Medium Minor

| Severity | Code | Description |
|---|---|---|
| | MC | Missing Check |
| | L01 | Public Function could be Declared External |
| | L02 | State Variables could be Declared Constant |
| | L04 | Conformance to Solidity Naming Conventions |
| | L07 | Missing Events Arithmetic |
| | L15 | Local Scope Variable Shadowing |

# MC - Missing Check

| | |
|---|---|
| **Criticality** | minor |
| **Location** | contract.sol#L188,222 |

## 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.

According to the ERC20 specification, the *transfer* and *transferFrom* methods return a boolean value that indicates if the transaction succeeded. If this value is not checked by the contract, then the method will proceed guessing that the recipient received the amount successfully even if it fails.

```
wRING.transferFrom(msg.sender, address(this), _amount);
//
wRING.transfer(nftOwner, transferAmount);
BUSD.transfer(nftOwner, lockInfo.busdReward);
```

## Recommendation

The contract should check if the return value of the *transfer* and *transferFrom* functions is true.

# L01 - Public Function could be Declared External

| | |
|---|---|
| **Criticality** | minor |
| **Location** | RingNote.sol#L84,99,168 |

## Description

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

```
remainingRING
claimBackDust
fillRewardBalances
```

## Recommendation

Use the external attribute for functions never called from the contract

# L02 - State Variables could be Declared Constant

| | |
|---|---|
| **Criticality** | minor |
| **Location** | RingNote.sol#L28 |

## Description

Constant state variables should be declared constant to save gas.

```
denominator
```

## Recommendation

Add the constant attribute to state variables that never change.

# L04 - Conformance to Solidity Naming Conventions

| | |
|---|---|
| **Criticality** | minor |
| **Location** | RingNote.sol#L84,186,23 |

## 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.

```
BUSD
_amount
_user
_busdAmount
```

## Recommendation

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

# L07 - Missing Events Arithmetic

| | |
|---|---|
| **Criticality** | minor |
| **Location** | RingNote.sol#L84 |

## Description

Detected missing events for critical arithmetic parameters. There are functions that have no event emitted, so it is difficult to track off-chain changes.

```
rewardBalance = rewardBalance.add(_busdAmount)
```

## Recommendation

Emit an event for critical parameter changes.

# L15 - Local Scope Variable Shadowing

| | |
|---|---|
| **Criticality** | minor |
| **Location** | RingNote.sol#L46,47,48 |

## Description

The are variables that are defined in the local scope containing the same name from an upper scope.

```
baseURI
symbol
name
```

## Recommendation

The local variables should have different names from the upper scoped variables.

# Contract Functions

| Contract | Type | Bases | | |
|---|---|---|---|---|
| | **Function Name** | **Visibility** | **Mutability** | **Modifiers** |
| | | | | |
| **Address** | Library | | | |
| | isContract | Internal | | |
| | sendValue | Internal | ✓ | |
| | functionCall | Internal | ✓ | |
| | functionCall | Internal | ✓ | |
| | functionCallWithValue | Internal | ✓ | |
| | functionCallWithValue | Internal | ✓ | |
| | functionStaticCall | Internal | | |
| | functionStaticCall | Internal | | |
| | functionDelegateCall | Internal | ✓ | |
| | functionDelegateCall | Internal | ✓ | |
| | _verifyCallResult | Private | | |
| | | | | |
| **Context** | Implementation | | | |
| | _msgSender | Internal | | |
| | _msgData | Internal | | |
| | | | | |
| **Counters** | Library | | | |
| | current | Internal | | |
| | increment | Internal | ✓ | |
| | decrement | Internal | ✓ | |
| | | | | |
| **EnumerableMap** | Library | | | |
| | _set | Private | ✓ | |
| | _remove | Private | ✓ | |
| | _contains | Private | | |
| | _length | Private | | |
| | _at | Private | | |

| | | | | |
|---|---|---|---|---|
| | _tryGet | Private | | |
| | _get | Private | | |
| | _get | Private | | |
| | set | Internal | ✓ | |
| | remove | Internal | ✓ | |
| | contains | Internal | | |
| | length | Internal | | |
| | at | Internal | | |
| | tryGet | Internal | | |
| | get | Internal | | |
| | get | Internal | | |
| | | | | |
| **EnumerableSet** | Library | | | |
| | _add | Private | ✓ | |
| | _remove | Private | ✓ | |
| | _contains | Private | | |
| | _length | Private | | |
| | _at | Private | | |
| | add | Internal | ✓ | |
| | remove | Internal | ✓ | |
| | contains | Internal | | |
| | length | Internal | | |
| | at | Internal | | |
| | add | Internal | ✓ | |
| | remove | Internal | ✓ | |
| | contains | Internal | | |
| | length | Internal | | |
| | at | Internal | | |
| | add | Internal | ✓ | |
| | remove | Internal | ✓ | |
| | contains | Internal | | |
| | length | Internal | | |
| | at | Internal | | |
| | | | | |
| **ERC165** | Implementation | IERC165 | | |

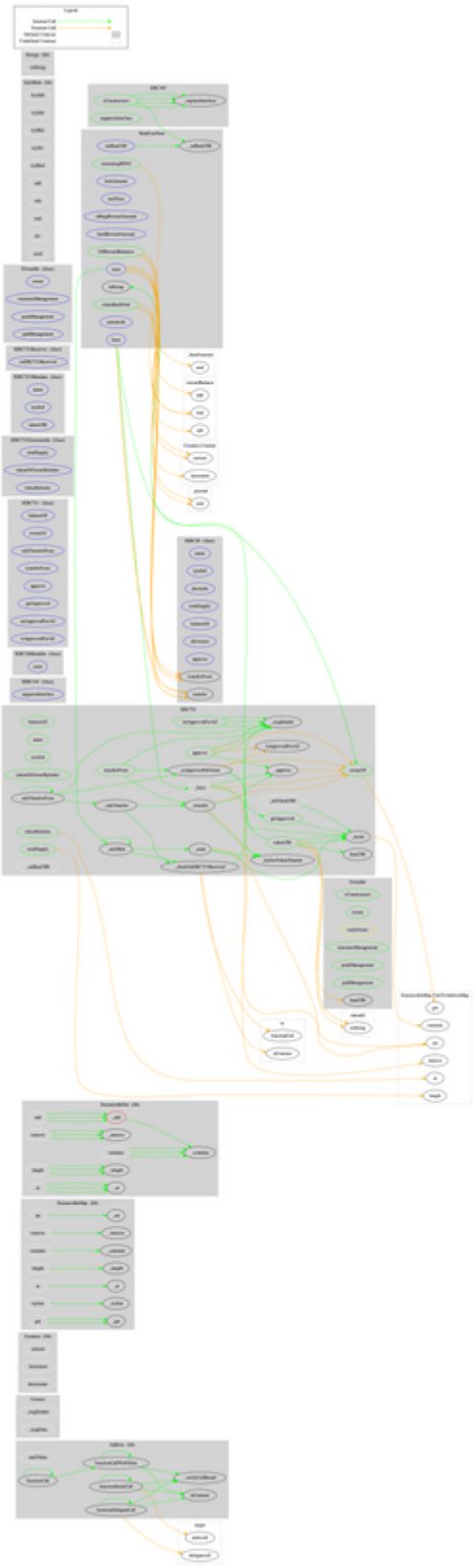
| | | | | |
|---|---|---|---|---|
| | <Constructor> | Internal | ✓ | |
| | supportsInterface | Public | | - |
| | _registerInterface | Internal | ✓ | |
| | | | | |
| **ERC721** | Implementation | Context, ERC165, IERC721, IERC721Metadata, IERC721Enumerable | | |
| | <Constructor> | Public | ✓ | - |
| | balanceOf | Public | | - |
| | ownerOf | Public | | - |
| | name | Public | | - |
| | symbol | Public | | - |
| | tokenURI | Public | | - |
| | baseURI | Public | | - |
| | tokenOfOwnerByIndex | Public | | - |
| | totalSupply | Public | | - |
| | tokenByIndex | Public | | - |
| | approve | Public | ✓ | - |
| | getApproved | Public | | - |
| | setApprovalForAll | Public | ✓ | - |
| | isApprovedForAll | Public | | - |
| | transferFrom | Public | ✓ | - |
| | safeTransferFrom | Public | ✓ | - |
| | safeTransferFrom | Public | ✓ | - |
| | _safeTransfer | Internal | ✓ | |
| | _exists | Internal | | |
| | _isApprovedOrOwner | Internal | | |
| | _safeMint | Internal | ✓ | |
| | _safeMint | Internal | ✓ | |
| | _mint | Internal | ✓ | |
| | _burn | Internal | ✓ | |
| | _transfer | Internal | ✓ | |
| | _setTokenURI | Internal | ✓ | |

| | | | | |
|---|---|---|---|---|
| | _setBaseURI | Internal | ✓ | |
| | _checkOnERC721Received | Private | ✓ | |
| | _approve | Internal | ✓ | |
| | _beforeTokenTransfer | Internal | ✓ | |
| | | | | |
| **IERC165** | Interface | | | |
| | supportsInterface | External | | - |
| | | | | |
| **IERC20** | Interface | | | |
| | name | External | | - |
| | symbol | External | | - |
| | decimals | External | | - |
| | totalSupply | External | | - |
| | balanceOf | External | | - |
| | allowance | External | | - |
| | approve | External | ✓ | - |
| | transfer | External | ✓ | - |
| | transferFrom | External | ✓ | - |
| | | | | |
| **IERC20Mintable** | Interface | | | |
| | mint | External | ✓ | - |
| | mint | External | ✓ | - |
| | | | | |
| **IERC721** | Interface | IERC165 | | |
| | balanceOf | External | | - |
| | ownerOf | External | | - |
| | safeTransferFrom | External | ✓ | - |
| | transferFrom | External | ✓ | - |
| | approve | External | ✓ | - |
| | getApproved | External | | - |
| | setApprovalForAll | External | ✓ | - |
| | isApprovedForAll | External | | - |
| | safeTransferFrom | External | ✓ | - |
| | | | | |

| | | | | |
|---|---|---|---|---|
| **IERC721Enumerable** | Interface | IERC721 | | |
| | totalSupply | External | | - |
| | tokenOfOwnerByIndex | External | | - |
| | tokenByIndex | External | | - |
| | | | | |
| **IERC721Metadata** | Interface | IERC721 | | |
| | name | External | | - |
| | symbol | External | | - |
| | tokenURI | External | | - |
| | | | | |
| **IERC721Receiver** | Interface | | | |
| | onERC721Received | External | ✓ | - |
| | | | | |
| **IOwnable** | Interface | | | |
| | owner | External | | - |
| | renounceManagement | External | ✓ | - |
| | pushManagement | External | ✓ | - |
| | pullManagement | External | ✓ | - |
| | | | | |
| **Ownable** | Implementation | IOwnable | | |
| | <Constructor> | Public | ✓ | - |
| | owner | Public | | - |
| | renounceManagement | Public | ✓ | onlyOwner |
| | pushManagement | Public | ✓ | onlyOwner |
| | pullManagement | Public | ✓ | - |
| | | | | |
| **RiskFreeNote** | Implementation | ERC721, Ownable | | |
| | <Constructor> | Public | ✓ | ERC721 |
| | fillRewardBalances | Public | ✓ | onlyOwner |
| | claimBackDust | Public | ✓ | onlyOwner |
| | tokenURI | Public | | - |
| | lockAmount | External | | - |
| | endTime | External | | - |

| | | | | |
|---|---|---|---|---|
| | wRingRewardAmount | External | | - |
| | busdRewardAmount | External | | - |
| | remainingRING | Public | | - |
| | setBaseURI | External | ✓ | onlyOwner |
| | mint | External | ✓ | - |
| | burn | External | ✓ | - |
| | unlockAll | External | ✓ | onlyOwner |
| | toString | Public | | - |
| | char | Public | | - |
| | | | | |
| **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 | | |
| | | | | |
| **Strings** | Library | | | |
| | toString | Internal | | |

# Contract Flow



# Summary

Ring Risk Free Note allows users to stack an amount and get the rewards when the stacked period elapsed. The users receive an NFT as a receipt for their staking transaction. The users burn the NFT when they want to claim their rewards. This audit focuses on the security aspects, code optimizations and business logic analysis.

# 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 provides all the essential tools to assist users draw their own conclusions.

The Cyberscope team

https://www.cyberscope.io