Cyberscope

Audit Report

Stream2Earn

August 2022

Table of Contents

Contract Review

Contract Name	STREAMN
Compiler Version	v0.8.10+commit.fc410830
Optimization	200 runs
Licence	None
Explorer	https://bscscan.com/token/0x3d051a2efbb7441e54c55285d983e86093780526
Symbol	STREAMN
Decimals	18
Total Supply	100,000,000
Domain	https://stream2earn.io

Source Files

Filename	**SHA256**
contract.sol	122c321fb0eb41b4867b002f73934115b69bc9bf8469e7aac39bfca74bb76afd

Audit Updates

Initial Audit	18th August 2022
Corrected	

Contract Analysis

● Critical   ● Medium   ● Minor   ● Pass

Severity	Code	Description
●	ST	Contract Owner is not able to stop or pause transactions
●	OCTD	Contract Owner is not able to transfer tokens from specific address
●	OTUT	Owner Transfer User's Tokens
●	ELFM	Contract Owner is not able to increase fees more than a reasonable percent (25%)
●	ULTW	Contract Owner is not able to increase the amount of liquidity taken by dev wallet more than a reasonable percent
●	MT	Contract Owner is not able to mint new tokens
●	BT	Contract Owner is not able to burn tokens from specific wallet
●	BC	Contract Owner is not able to blacklist wallets from selling

Contract Diagnostics

● Critical   ● Medium   ● Minor

Severity	Code	Description
●	FSA	Fixed Swap Address
●	CO	Code Optimization
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L04	Conformance to Solidity Naming Conventions
●	L05	Unused State Variable

FSA - Fixed Swap Address

Criticality	minor
Location	contract.sol#L263

Description

The swap address is assigned once in the constructor and it can not be changed. The decentralized swaps sometimes create a new swap version or abandon the current. A contract that cannot change the swap address may not be able to catch-up the upgrade.

```
constructor() {
    _owned[_msgSender()] = _rTotal;
    IUniswapV2Router02 _uniswapV2Router = IUniswapV2Router02(
        0x10ED43C718714eb63d5aA57B78B54704E256024E
    );
    uniswapV2Router = _uniswapV2Router;
    uniswapV2Pair = IUniswapV2Factory(_uniswapV2Router.factory())
        .createPair(address(this), _uniswapV2Router.WETH());
```

Recommendation

It could be better to allow the swap address mutation in case of future swap updates.

CO - Code Optimization

Criticality	minor
Location	contract.sol#L535,546

Description

There are code segments that could be optimized. A segment may be optimized so that it becomes a smaller size, consumes less memory, executes more rapidly, or performs fewer operations.

The _tokenTransfer method is redundant.

```
function _tokenTransfer(
    address sender,
    address recipient,
    uint256 amount
) private {
    _transferStandard(sender, recipient, amount);
}


function _transferStandard(
    address sender,
    address recipient,
    uint256 tAmount
) private {
```

Recommendation

The method _tokenTransfer can be merged with _transferStandard method.

Rewrite some code segments so the runtime will be more performant.

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L332,151,161,388,363,678,344,685

Description

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

```
excludeFromFees
allowance
toggleSwap
approve
transferFrom
transferOwnership
renounceOwnership
transfer
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract.sol#L134,219,212,210,221,211

Description

Constant state variables should be declared constant to save gas.

```
_symbol
_charityAddress
_name
_decimals
_marketingAddress
_previousOwner
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L678,62

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.

```
WETH
_feeSwap
```

Recommendation

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

L05 - Unused State Variable

Criticality	minor
Location	contract.sol#L134

Description

There are segments that contain unused state variables.

_previousOwner

Recommendation

Remove unused state variables.

Contract Functions

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
Token	Interface			
	transferFrom	External	✓	-
	transfer	External	✓	-
IUniswapV2Factory	Interface			
	createPair	External	✓	-
IUniswapV2Router02	Interface			
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
	factory	External		-
	WETH	External		-
	addLiquidityETH	External	Payable	-
Context	Implementation			
	_msgSender	Internal		
SafeMath	Library			
	add	Internal		
	sub	Internal		

	sub	Internal		
	mul	Internal		
	div	Internal		
	div	Internal		
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
IERC20Metadata	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	External		-
STREAMN	Implementation	Context, IERC20, Ownable, IERC20Metadata		
	<Constructor>	Public	✓	-
	decimals	External		-
	symbol	External		-
	name	External		-
	totalSupply	External		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	tokenFromReflection	Private		
	_approve	Private	✓	
	_transfer	Private	✓	
	swapTokensForEth	Private	✓	lockTheSwap
	sendETHToFee	Private	✓	

	_tokenTransfer	Private	✓	
	_transferStandard	Private	✓	
	_takeTeam	Private	✓	
	_reflectFee	Private	✓	
	<Receive Ether>	External	Payable	-
	_getValues	Private		
	_getTValues	Private		
	_getRValues	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	toggleSwap	Public	✓	onlyDev
	excludeFromFees	Public	✓	onlyDev
	burn	External	✓	-
	_burn	Internal	✓	
	burnFrom	External	✓	-

Contract Flow
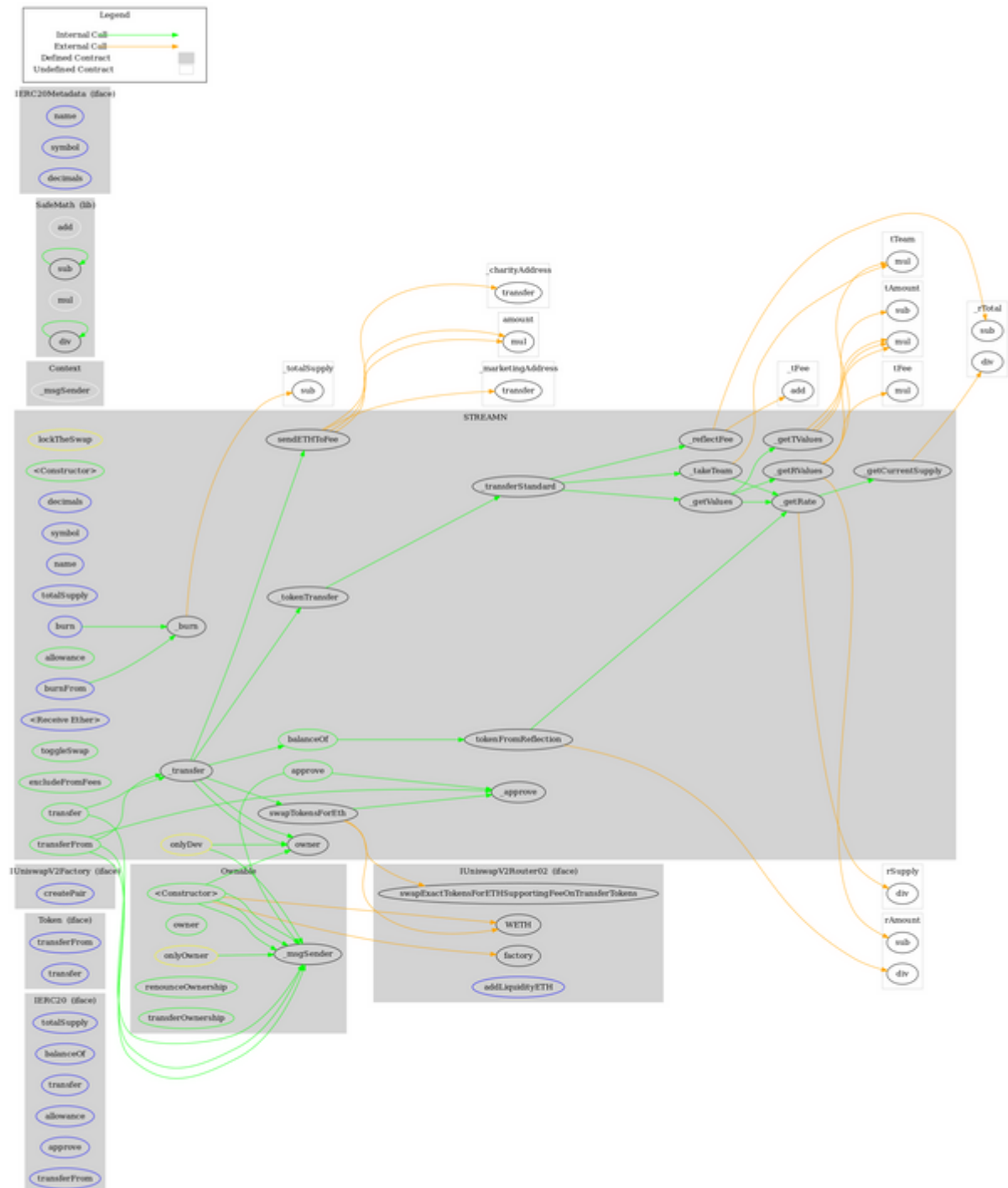
Domain Info

Domain Name	stream2earn.io
Registry Domain ID	ac08ad14b8f249df8f520f9ff78440b3-DONUTS
Creation Date	2022-08-14T15:38:25Z
Updated Date	2022-08-14T15:38:26Z
Registry Expiry Date	2023-08-14T15:38:25Z
Registrar WHOIS Server	whois.godaddy.com/
Registrar URL	http://www.godaddy.com/domains/search.aspx?ci=8990
Registrar	GoDaddy.com, LLC
Registrar IANA ID	146

The domain has been created in 12 months before the creation of the audit.

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

Summary

Stream2Earn is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler error or critical issues. The contract Owner can access some admin functions that can not be used in a malicious way to disturb the users' transactions. There is also a limit of max 3% fees.

Summary

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