Cyberscope

Audit Report

Shib2.0

April 2022

Type	BEP20
Network	BSC
Address	0xDF0825E5F9B37b9C2c560c0771A0a2118f8d78b0
Audited by	© cyberscope

Table of Contents

Table of Contents 1
Contract Review 3
Audit Updates 3
Source Files 4
Contract Analysis 6
Contract Diagnostics 7
L01 - Public Function could be Declared External 8
Description 8
Recommendation 8
L02 - State Variables could be Declared Constant 9
Description 9
Recommendation 9
L04 - Conformance to Solidity Naming Conventions 10
Description 10
Recommendation 10
L07 - Missing Events Arithmetic 11
Description 11
Recommendation 11
L09 - Dead Code Elimination 12
Description 12
Recommendation 12
L12 - Using Variables before Declaration 13
Description 13
Recommendation 13
L14 - Uninitialized Variables in Local Scope 14
Description 14

Recommendation 14
L15 - Local Scope Variable Shadowing 15
Description 15
Recommendation 15
Contract Functions 16
Contract Flow 23
Domain Info 24
Summary 25
Disclaimer 26
About Cyberscope 27

Contract Review

Contract Name	Token
Compiler Version	v0.6.12+commit.27d51765
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0xdf0825e5f9b37b9c2c560c0771a0a2118f8d78b0
Symbol	Shib2.0
Decimals	9
Total Supply	100,000,000,000,000,000
Domain	shib20.co

Audit Updates

Initial Audit	25th April 2022
Corrected	

Source Files

Filename	SHA256
Context.sol	89affd14d10a0b91c1e1a1a904edfa058a7fadbace53545967bd2d0eae7684ec
DividendPayingToken.sol	4e63996cf6a3d0952d597eae65a516c50ff23ad1cd29dcde6296fc1a7403a8d9
DividendPayingTokenInterface.sol	edfb5ad20fed899a19ae93b7b4ef52b4a38e2eec6fd5a634bc07c5685c1762a1
DividendPayingTokenOptionalInterface.sol	fb9de2f6921d983fc09a004b8c6bc79c95642331cac36380301398af65dce6a7
ERC20.sol	1a121f91a91c5768a782536cfab68c0decce103cfe18898835f44610d1f0fc05
IERC20.sol	0f6a9850e975d5042c2f4f8f150df356b64f01358184396a99bb900a6d87ea55
IERC20Metadata.sol	07b046873251c771868686065501ac0c082926ec1306d4ad254094c6cd99a36b
IterableMapping.sol	971ad4c5959bea1ca1a19742bbac93bace7e311d77faf5631891dfb4d41e37ac
IUniswapV2Factory.sol	c780b51573992000d5ee28b54f8f8bd1b7e748791241b6a1486e36ba837218a7
IUniswapV2Pair.sol	bcc1435c9c6a949a7636555c4e2a4757ece265c3214fe04f00b27d23195fd22e
IUniswapV2Router.sol	cd94ad2ebf5f2bf1fbd07b280076e834696425c12bce50a9ba21a5384703e3bb
Ownable.sol	53847025cfbc9b20f9927c982b779445d6951157601744d81d9ccd7e81351e5a
SafeMath.sol	b3884a8ad3e218e4cc4267f029ce809e938b22f9eece2f4ffdc662bc277f14b1

SafeMathInt.sol	a3c0a178b8ff244c921b6951f4bd71b446e8ed08143718cc0ce5a08eb7ccd63c
SafeMathUint.sol	e5f809f812139641fdaecf33d48aafb4c0aea5368c875de375cb4256b613cb05
Token.sol	afc29765bbf9a5ed557fd8c9656359207dad331604ebda3453ba80a0ce398b99

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
	L01	Public Function could be Declared External
	L02	State Variables could be Declared Constant
	L04	Conformance to Solidity Naming Conventions
	L07	Missing Events Arithmetic
	L09	Dead Code Elimination
	L12	Using Variables before Declaration
	L14	Uninitialized Variables in Local Scope
	L15	Local Scope Variable Shadowing

L01 - Public Function could be Declared External

Criticality	minor
Location	DividendPayingToken.sol#L51,66,100,118 Token.sol#L128,153,183,213,238,242,250,625,684

Description

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

```
process
getAccountAtIndex
dividendTokenBalanceOf
withdrawableDividendOf
isExcludedFromFees
updateGasForProcessing
setAutomatedMarketMakerPair
updateUniswapV2Router
updateDividendTracker
...
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality	minor
Location	DividendPayingToken.sol#L21 Token.sol#L23,28,36,22,29,30

Description

Constant state variables should be declared constant to save gas.

```
marketingFee
liquidityFee
deadWallet
_liquidityReceiver
ShibRewardsFee
Shib
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	DividendPayingToken.sol#L100,107,118,127,21,26 Token.sol#L23,26,28,34,36,576

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.

```
_account
_liquidityReceiver
_marketingAddress
ShibRewardsFee
_isBot
Shib
magnitude
_owner
...
```

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	Token.sol#L179

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.

```
swapTokensAtAmount = value
```

Recommendation

Emit an event for critical parameter changes.

L09 - Dead Code Elimination

Criticality	minor
Location	DividendPayingToken.sol#L145

Description

Functions that are not used in the contract, and make the code's size bigger.

```
_transfer
```

Recommendation

Remove unused functions.

L12 - Using Variables before Declaration

Criticality	minor
Location	Token.sol#L391,393,392

Description

The contract is using a variable before the declaration. This is usually happening either if it has not been declared yet or the variable has been declared in a different scope.

```
claims
lastProcessedIndex
iterations
```

Recommendation

The variables should be declared before any usage of them.

L14 - Uninitialized Variables in Local Scope

Criticality	minor
Location	Token.sol#L392,391,393

Description

The are variables that are defined in the local scope and are not initialized.

```
lastProcessedIndex
iterations
claims
```

Recommendation

All the local scoped variables should be initialized.

L15 - Local Scope Variable Shadowing

Criticality	minor
Location	DividendPayingToken.sol#L46,100,107,118,127

Description

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

```
_owner
_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**
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
DividendPayingToken	Implementation	ERC20, Ownable, DividendPayingTokenInterface		
	<Constructor>	Public	✓	ERC20
	distributeShibDividends	Public	✓	onlyOwner
	withdrawDividend	Public	✓	-
	_withdrawDividendOfUser	Internal	✓	
	dividendOf	Public		-
	withdrawableDividendOf	Public		-
	withdrawnDividendOf	Public		-
	accumulativeDividendOf	Public		-
	_transfer	Internal	✓	
	_mint	Internal	✓	
	_burn	Internal	✓	
	_setBalance	Internal	✓	
DividendPayingTokenInterface	Interface			
	dividendOf	External		-
	withdrawDividend	External	✓	-
DividendPayingTokenOptionalInterface	Interface			

	dividendOf	External		-
	withdrawDividend	External	✓	-
ERC20	Implementation	Context, IERC20, IERC20Metadata		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	_transfer	Internal	✓	
	_mint	Internal	✓	
	_burn	Internal	✓	
	_approve	Internal	✓	
	_beforeTokenTransfer	Internal	✓	
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IERC20Metadata	Interface	IERC20		
	name	External		-
	symbol	External		-

	decimals	External		-
IterableMapping	Library			
	get	Public		-
	getIndexOfKey	Public		-
	getKeyAtIndex	Public		-
	size	Public		-
	set	Public	✓	-
	remove	Public	✓	-
IUniswapV2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
IUniswapV2Pair	Interface			
	name	External		-
	symbol	External		-
	decimals	External		-
	totalSupply	External		-
	balanceOf	External		-
	allowance	External		-
	approve	External	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-
	DOMAIN_SEPARATOR	External		-
	PERMIT_TYPEHASH	External		-
	nonces	External		-

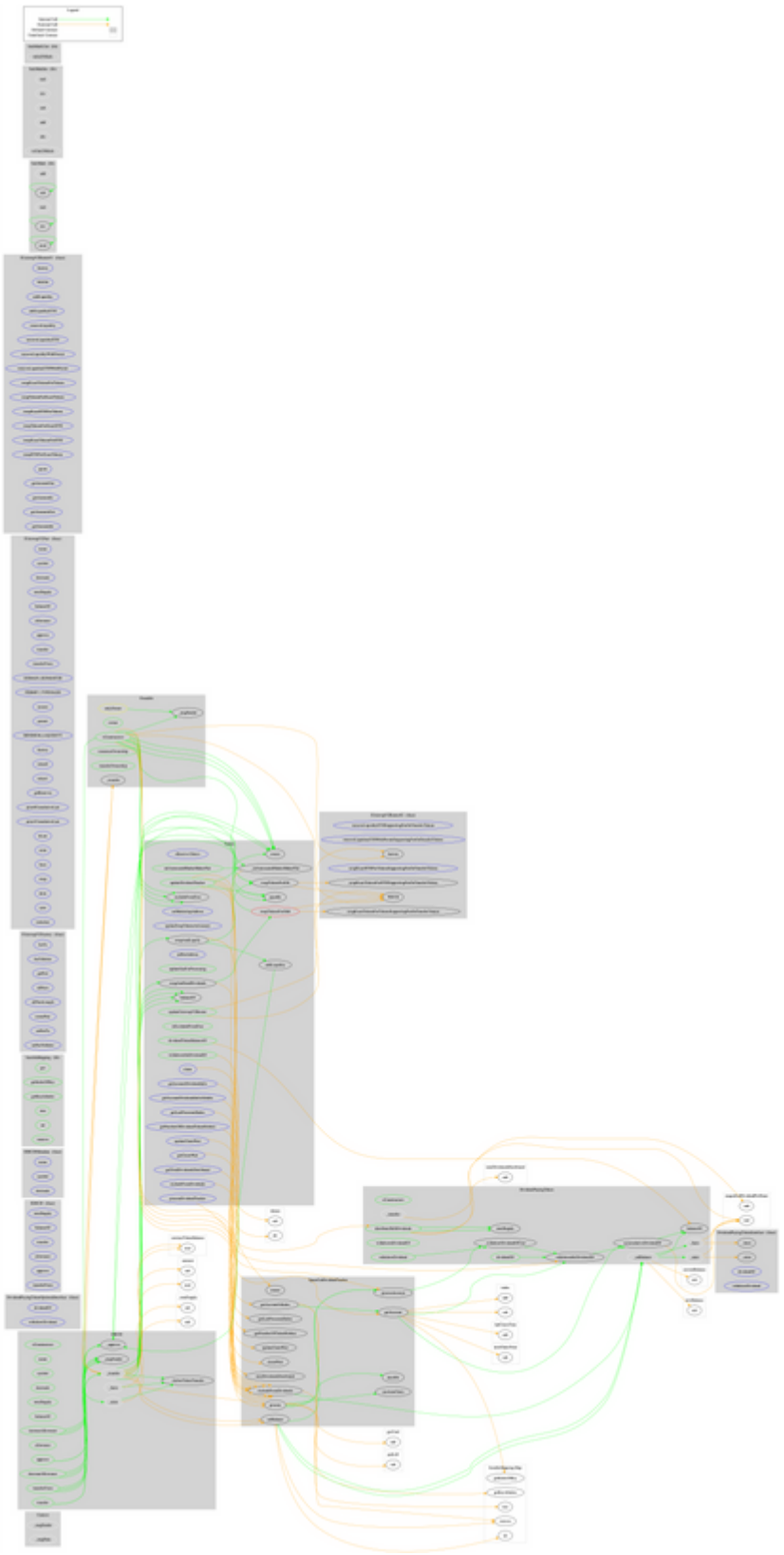
	permit	External	✓	-
	MINIMUM_LIQUIDITY	External		-
	factory	External		-
	token0	External		-
	token1	External		-
	getReserves	External		-
	price0CumulativeLast	External		-
	price1CumulativeLast	External		-
	kLast	External		-
	mint	External	✓	-
	burn	External	✓	-
	swap	External	✓	-
	skim	External	✓	-
	sync	External	✓	-
	initialize	External	✓	-
IUniswapV2Router01	Interface			
	factory	External		-
	WETH	External		-
	addLiquidity	External	✓	-
	addLiquidityETH	External	Payable	-
	removeLiquidity	External	✓	-
	removeLiquidityETH	External	✓	-
	removeLiquidityWithPermit	External	✓	-
	removeLiquidityETHWithPermit	External	✓	-
	swapExactTokensForTokens	External	✓	-
	swapTokensForExactTokens	External	✓	-
	swapExactETHForTokens	External	Payable	-
	swapTokensForExactETH	External	✓	-
	swapExactTokensForETH	External	✓	-
	swapETHForExactTokens	External	Payable	-
	quote	External		-
	getAmountOut	External		-
	getAmountIn	External		-
	getAmountsOut	External		-

	getAmountsIn	External		-
IUniswapV2Router02	Interface	IUniswapV2Router01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
SafeMath	Library			
	add	Internal		
	sub	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	div	Internal		
	mod	Internal		
	mod	Internal		
SafeMathInt	Library			
	mul	Internal		
	div	Internal		
	sub	Internal		
	add	Internal		
	abs	Internal		

	toUint256Safe	Internal		
SafeMathUint	Library			
	toInt256Safe	Internal		
Token	Implementation	ERC20, Ownable		
	<Constructor>	Public	✓	ERC20
	<Receive Ether>	External	Payable	-
	updateDividendTracker	Public	✓	onlyOwner
	updateUniswapV2Router	Public	✓	onlyOwner
	excludeFromFees	Public	✓	onlyOwner
	setMarketingAddress	External	✓	onlyOwner
	updateSwapTokensAtAmount	External	✓	onlyOwner
	setAutomatedMarketMakerPair	Public	✓	onlyOwner
	setBotAddress	External	✓	onlyOwner
	_setAutomatedMarketMakerPair	Private	✓	
	updateGasForProcessing	Public	✓	onlyOwner
	updateClaimWait	External	✓	onlyOwner
	getClaimWait	External		-
	getTotalDividendsDistributed	External		-
	isExcludedFromFees	Public		-
	withdrawableDividendOf	Public		-
	dividendTokenBalanceOf	Public		-
	excludeFromDividends	External	✓	onlyOwner
	getAccountDividendsInfo	External		-
	getAccountDividendsInfoAtIndex	External		-
	processDividendTracker	External	✓	-
	claim	External	✓	-
	getLastProcessedIndex	External		-
	getNumberOfDividendTokenHolders	External		-
	_transfer	Internal	✓	
	swapAndLiquify	Private	✓	
	swapTokensForEth	Private	✓	
	swapTokensForShib	Private	✓	
	addLiquidity	Private	✓	

	swapAndSendDividends	Private	✓	
SpaceCatDividendTracker	Implementation	Ownable, DividendPayingToken		
	<Constructor>	Public	✓	DividendPayingToken
	_transfer	Internal	✓	
	withdrawDividend	Public	✓	-
	excludeFromDividends	External	✓	onlyOwner
	updateClaimWait	External	✓	onlyOwner
	getLastProcessedIndex	External		-
	getNumberOfTokenHolders	External		-
	getAccount	Public		-
	getAccountAtIndex	Public		-
	canAutoClaim	Private		
	setBalance	External	✓	onlyOwner
	process	Public	✓	-
	processAccount	Public	✓	onlyOwner

Contract Flow



Domain Info

Domain Name	shib20.co
Registry Domain ID	D57E3D191D346474C9B496621DB9C85F7-GDREG
Creation Date	2022-04-24T09:09:22Z
Updated Date	2022-04-24T09:09:25Z
Registry Expiry Date	2023-04-24T09:09:22Z
Registrar WHOIS Server	whois.namecheap.com
Registrar URL	http://www.namecheap.com
Registrar	NameCheap, Inc.
Registrar IANA ID	1068

The domain has been created 1 day before the creation of the audit. It will expire in 12 months.

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

Summary

Shib2.0 is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler error or critical issues. The contract ownership is renounced, thus the owner cannot interact with the owner’s methods. The fees are fixed to 10%.

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