



Cyberscope

Audit Report

Fitness Token

July 2022

Type BEP20

Network BSC

Address 0xfd77214b4a1aa6f962972cbe65fb179c501d9094

Audited by © cyberscope

Table of Contents

Table of Contents	1
Contract Review	3
Source Files	3
Audit Updates	3
Contract Analysis	4
OCTD - Owner Contract Tokens Drain	5
Description	5
Recommendation	5
ULTW - Unlimited Liquidity to Team Wallet	6
Description	6
Recommendation	6
MT - Mint Tokens	7
Description	7
Recommendation	7
BT - Burn Tokens	8
Description	8
Recommendation	8
Contract Diagnostics	9
FSA - Fixed Swap Address	10
Description	10
Recommendation	10
L01 - Public Function could be Declared External	11
Description	11
Recommendation	11
L02 - State Variables could be Declared Constant	12
Description	12

Recommendation	12
L04 - Conformance to Solidity Naming Conventions	13
Description	13
Recommendation	13
L09 - Dead Code Elimination	14
Description	14
Recommendation	14
L14 - Uninitialized Variables in Local Scope	15
Description	15
Recommendation	15
Contract Functions	16
Contract Flow	21
Domain Info	22
Summary	23
Disclaimer	24
About Cyberscope	25

Contract Review

Contract Name	SecondToken
Compiler Version	v0.8.12+commit.f00d7308
Optimization	20 runs
Licence	MIT
Explorer	https://bscscan.com/token/0xfd77214b4a1aa6f962972cbe65fb179c501d9094
Symbol	FIT
Decimals	18
Total Supply	-
Domain	https://healthfi.app/

Source Files

Filename	SHA256
contract.sol	7c9049b44dbe5167e13c03a9814cec3fac5e9822c5bac3c5dbce7bfd75eaa898

Audit Updates

Initial Audit	7th July 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

OCTD - Owner Contract Tokens Drain

Criticality	minor
Location	contract.sol#L1518

Description

The contract owner has the authority to claim all the balance of the contract. The owner may take advantage of it by calling the `sweepTokenForBosses` function.

```
function sweepTokenForBosses(uint256 tokenAmount)
    external
    onlyRole(SWEEPER_ROLE)
{
    uint256 contractTokenBalance = balanceOf(address(this));
    if (tokenAmount > contractTokenBalance) {
        tokenAmount = contractTokenBalance;
    }
    if (tokenAmount >= tokenForBosses) {
        _swapTokensForEth(tokenForBosses);
    } else {
        _swapTokensForEth(tokenAmount);
    }
}
```

Recommendation

The team should carefully manage the private keys of the owner's account. We strongly recommend a powerful security mechanism that will prevent a single user from accessing the contract admin functions. That risk can be prevented by temporarily locking the contract or renouncing ownership.

ULTW - Unlimited Liquidity to Team Wallet

Criticality	minor
Location	contract.sol#L1533

Description

The contract owner has the authority to transfer funds without limit to the team wallet. These funds have been accumulated from fees collected from the contract. The owner may take advantage of it by calling the `withdraw` method.

```
function withdraw() external onlyRole(BOSS_CONTROL_ROLE) {  
    // just in-case swap not work  
    uint256 contractTokenBalance = balanceOf(address(this));  
    _transfer(address(this), addressForBosses, contractTokenBalance);  
}
```

Recommendation

The contract could embody a check for the maximum amount of funds that can be swapped. Since a huge amount may volatile the token's price.

The team should carefully manage the private keys of the owner's account. We strongly recommend a powerful security mechanism that will prevent a single user from accessing the contract admin functions. That risk can be prevented by temporarily locking the contract or renouncing ownership.

MT - Mint Tokens

Criticality	critical
Location	contract.sol#L1504

Description

The contract owner has the authority to mint tokens. The owner may take advantage of it by calling the `mint` function. As a result the contract tokens will be highly inflated.

```
function mint(address account, uint256 amount)
    external
    onlyRole(MINTER_ROLE)
{
    _mint(account, amount);
}
```

Recommendation

The owner should carefully manage the credentials of the owner's account. We advised considering an extra-strong security mechanism that the actions may be quarantined by many users instead of one. The owner could also renounce the contract ownership for a period of time or pass the access to the zero address.

BT - Burn Tokens

Criticality	critical
Location	contract.sol#L1511

Description

The contract owner has the authority to burn tokens from a specific address. The owner may take advantage of it by calling the `burn` function. As a result the targeted contract address will lose the corresponding tokens.

```
function burn(address account, uint256 amount)
    external
    onlyRole(BURNER_ROLE)
{
    _burn(account, amount);
}
```

Recommendation

The owner should carefully manage the credentials of the owner's account. We advised considering an extra-strong security mechanism that the actions may be quarantined by many users instead of one. The owner could also renounce the contract ownership for a period of time or pass the access to the zero address.

Contract Diagnostics

● Critical ● Medium ● Minor

Severity	Code	Description
●	FSA	Fixed Swap Address
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L04	Conformance to Solidity Naming Conventions
●	L09	Dead Code Elimination
●	L14	Uninitialized Variables in Local Scope

FSA - Fixed Swap Address

Criticality	minor
Location	contract.sol#L1598

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.

```
Fitness Token
IUniswapV2Router02 _uniswapV2Router = IUniswapV2Router02(
    newRouter // address uniswap
);
uniswapV2Pair = IUniswapV2Factory(_uniswapV2Router.factory())
    .createPair(address(this), _uniswapV2Router.WETH());
uniswapV2Router = _uniswapV2Router;
```

Recommendation

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

L01 - Public Function could be Declared External

Criticality

minor

Location

contract.sol#L529,458,572,441,484,552,433,1116,1129,465,507,1147

Description

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

```
renounceRole
approve
totalSupply
revokeRole
grantRole
name
increaseAllowance
transfer
symbol
...
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality

minor

Location

contract.sol#L1482

Description

Constant state variables should be declared constant to save gas.

```
maxSupply
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L1421,1448,1408,1434,1217,1398

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.

```
_target  
WETH  
_tokenForBosses  
_sellFeeRate  
_buyFeeRate  
_addressForBosses
```

Recommendation

Follow the Solidity naming convention.

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

L09 - Dead Code Elimination

Criticality

minor

Location

contract.sol#L1180,888,863

Description

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

```
toString  
toHexString  
_setRoleAdmin
```

Recommendation

Remove unused functions.

L14 - Uninitialized Variables in Local Scope

Criticality

minor

Location

contract.sol#L1557

Description

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

```
transferFeeRate
```

Recommendation

All the local scoped variables should be initialized.

Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
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		
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IERC20Metadata	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	External		-

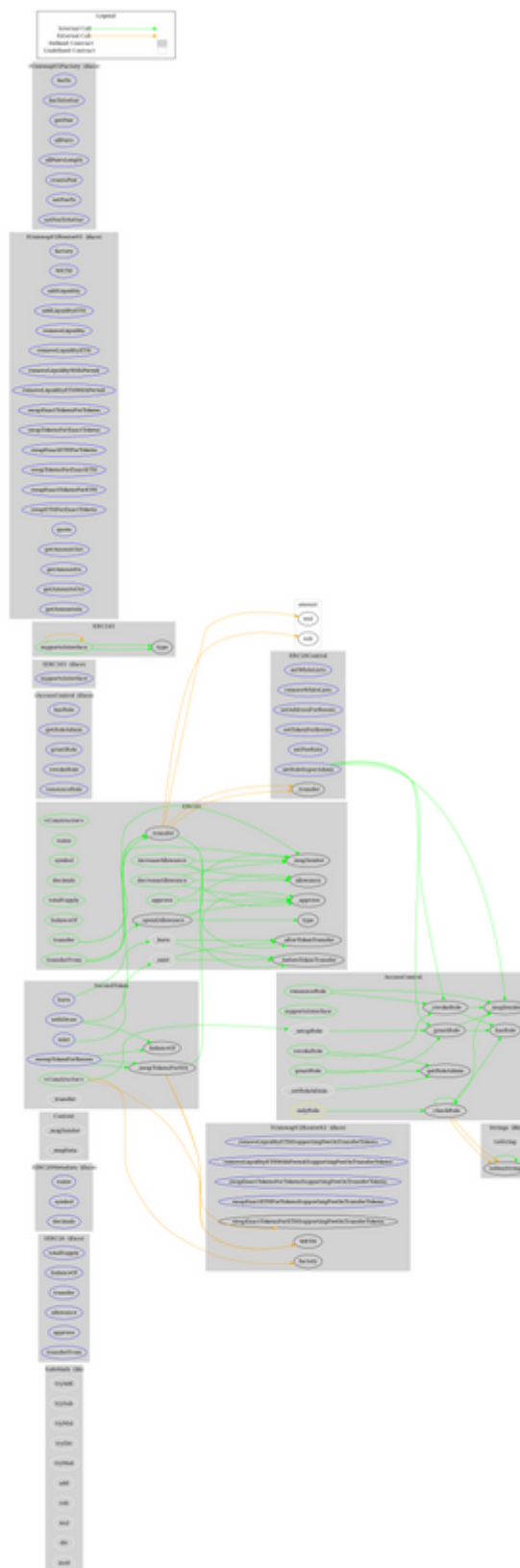
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
ERC20	Implementation	Context, IERC20, IERC20Met adata		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	_transfer	Internal	✓	
	_mint	Internal	✓	
	_burn	Internal	✓	
	_approve	Internal	✓	
	_spendAllowance	Internal	✓	
	_beforeTokenTransfer	Internal	✓	
	_afterTokenTransfer	Internal	✓	
IAccessControl	Interface			
	hasRole	External		-
	getRoleAdmin	External		-
	grantRole	External	✓	-
	revokeRole	External	✓	-
	renounceRole	External	✓	-

Strings	Library			
	toString	Internal		
	toHexString	Internal		
	toHexString	Internal		
IERC165	Interface			
	supportsInterface	External		-
ERC165	Implementation	IERC165		
	supportsInterface	Public		-
AccessControl	Implementation	Context, IAccessControl, ERC165		
	supportsInterface	Public		-
	hasRole	Public		-
	_checkRole	Internal		
	_checkRole	Internal		
	getRoleAdmin	Public		-
	grantRole	Public	✓	onlyRole
	revokeRole	Public	✓	onlyRole
	renounceRole	Public	✓	-
	_setupRole	Internal	✓	
	_setRoleAdmin	Internal	✓	
	_grantRole	Internal	✓	
	_revokeRole	Internal	✓	
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	✓	-
IUniswapV2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-

ERC20Control	Implementation	AccessControl		
	setWhiteLists	External	✓	onlyRole
	removeWhiteLists	External	✓	onlyRole
	setAddressForBosses	External	✓	onlyRole
	setTokenForBosses	External	✓	onlyRole
	setFeeRate	External	✓	onlyRole
	setRoleSuperAdmin	External	✓	onlyRole
SecondToken	Implementation	ERC20, ERC20Control		
	<Constructor>	Public	✓	ERC20
	mint	External	✓	onlyRole
	burn	External	✓	onlyRole
	sweepTokenForBosses	External	✓	onlyRole
	withDraw	External	✓	onlyRole
	_swapTokensForEth	Private	✓	
	_transfer	Internal	✓	

Contract Flow



Domain Info

Domain Name	healthfi.app
Registry Domain ID	48E38B29B-APP
Creation Date	2022-05-03T12:10:51Z
Updated Date	2022-05-08T12:10:51Z
Registry Expiry Date	2024-05-03T12:10:51Z
Registrar WHOIS Server	whois.godaddy.com
Registrar URL	https://www.godaddy.com/
Registrar	GoDaddy.com, LLC
Registrar IANA ID	146

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

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

Summary

There are some functions that can be abused by the owner like transferring tokens to the team's wallet, transferring funds to the team's wallet, minting tokens and burning tokens. if the contract owner abuse the mint functionality, then the contract will be highly inflated. The users could lost their tokens. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats. There is also a limit of max 20% fees.

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>