Cyberscope

Audit Report

SpaceFalconX

June 2022

Type	BEP20
Network	BSC
Address	0xDC3a56E9A67BD212dFA870d1018e53489521706e
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	SpaceFalconX
Compiler Version	v0.8.13+commit.abaa5c0e
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0xDC3a56E9A67BD212dFA870d1018e53489521706e
Symbol	$SFX
Decimals	18
Total Supply	1,000,000,000,000
Domain	spacefalconx.com

Source Files

Filename	**SHA256**
contract.sol	89954ca3891073640deac3633f5e6a3eeeeaa7de90c6f7b57aafb6b1f2f70764

Audit Updates

Initial Audit	6th June 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#L762

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 `rescueStuckBNB` or `rescueBEP20` function.

```
function rescueStuckBNB() external onlyAuthorized {
    uint256 bnbAmount = address(this).balance;
    payable(msg.sender).transfer(bnbAmount);
}

function rescueBEP20(address _token) external onlyAuthorized {
    uint256 tamt = IERC20(_token).balanceOf(address(this));
    IERC20(_token).transfer(msg.sender, tamt);
}
```

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.

ELFM - Exceed Limit Fees Manipulation

Criticality	medium
Location	contract.sol#L625

Description

The contract owner has the authority to increase over the allowed limit of 25%. The owner may take advantage of it by calling the `setFees` function with the maximum allowed values. As a result the sale transactions will be taxed by `5 + 5 + 5 + 5 + 5 + 6 = 31%`

```
function setFees(uint256 marketingfee, uint256 devfee, uint256 treasuryfee,
uint256 liquidityfee, uint256 rewardsfee, uint256 sellfeeIncrease) external
onlyOwner{
    require(rewardsfee <= 5, "Requested rewardsFee fee not within acceptable
range.");
    require(liquidityfee <= 5 , "Requested liquidity fee not within acceptable
range.");
    require(marketingfee <= 5, "Requested marketing fee not within acceptable
range.");
    require(devfee <= 5, "Requested marketing fee not within acceptable
range.");
    require(treasuryfee <= 5, "Requested marketing fee not within acceptable
range.");
    require(sellfeeIncrease <= 6, "Requested sell fee increase not within
acceptable range.");
    rewardsFee = rewardsfee;
    liquidityFee = liquidityfee;
    marketingFee = marketingfee;
    devFee = devfee;
    treasuryFee = treasuryfee;
    sellFeeIncrease = sellfeeIncrease;
    totalFees = rewardsfee + liquidityfee + marketingfee + treasuryfee + devfee;
    emit SetFees(rewardsfee, liquidityfee, marketingfee);
}
```

Recommendation

The contract could embody a check for the maximum acceptable value.

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.

Contract Diagnostics

Critical Medium Minor

Severity	Code	Description
	CO	Code Optimization
	MC	Missing Check
	L04	Conformance to Solidity Naming Conventions

CO - Code Optimization

Criticality	minor
Location	contract.sol#L748

Description

The `burnAmt` is already checked in the setter method (`planBurn()`). As a result, the `require(..)` expression will never yield false during the runtime.

```
function doBurn(uint256 burnAmt) internal inburn {
    require(burnAmt <= 50 * TOTAL_SUPPLY / 1000, "burnAmount is limited to 5% in
single transaction");
    super._transfer(address(this), deadAddress, burnAmt);
    accumulatingForBurn = false;
}
```

Recommendation

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

MC - Missing Check

Criticality	medium
Location	contract.sol#L773

Description

The property `totalFees` is used as a divider in the `swapTokens()` method. The `totalFees` can be configured with zero value in the `setFees()` method. If the contract owner sets the `totalFees` to zero, then the transactions will revert. Since the `swapTokens` is called on the sales, then the contract can be converted into a **honeypot**.

```
function swapTokens(uint256 tokens) internal inSwap {
    uint256 LPtokens = tokens * liquidityFee / totalFees / 2;
    ....
```

Recommendation

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

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L19,82,66,110,112,114,116,117,118,200,201,210,212,224,530,605,767,398,399,400

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.

```
deadAddress
_maxWalletToken
_maxTxAmount
_token
_minDistribution
_minPeriod
_tradeCooldown
dividendsPerShareAccuracyFactor
_uniswapV2Router
...
```

Recommendation

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

Contract Functions

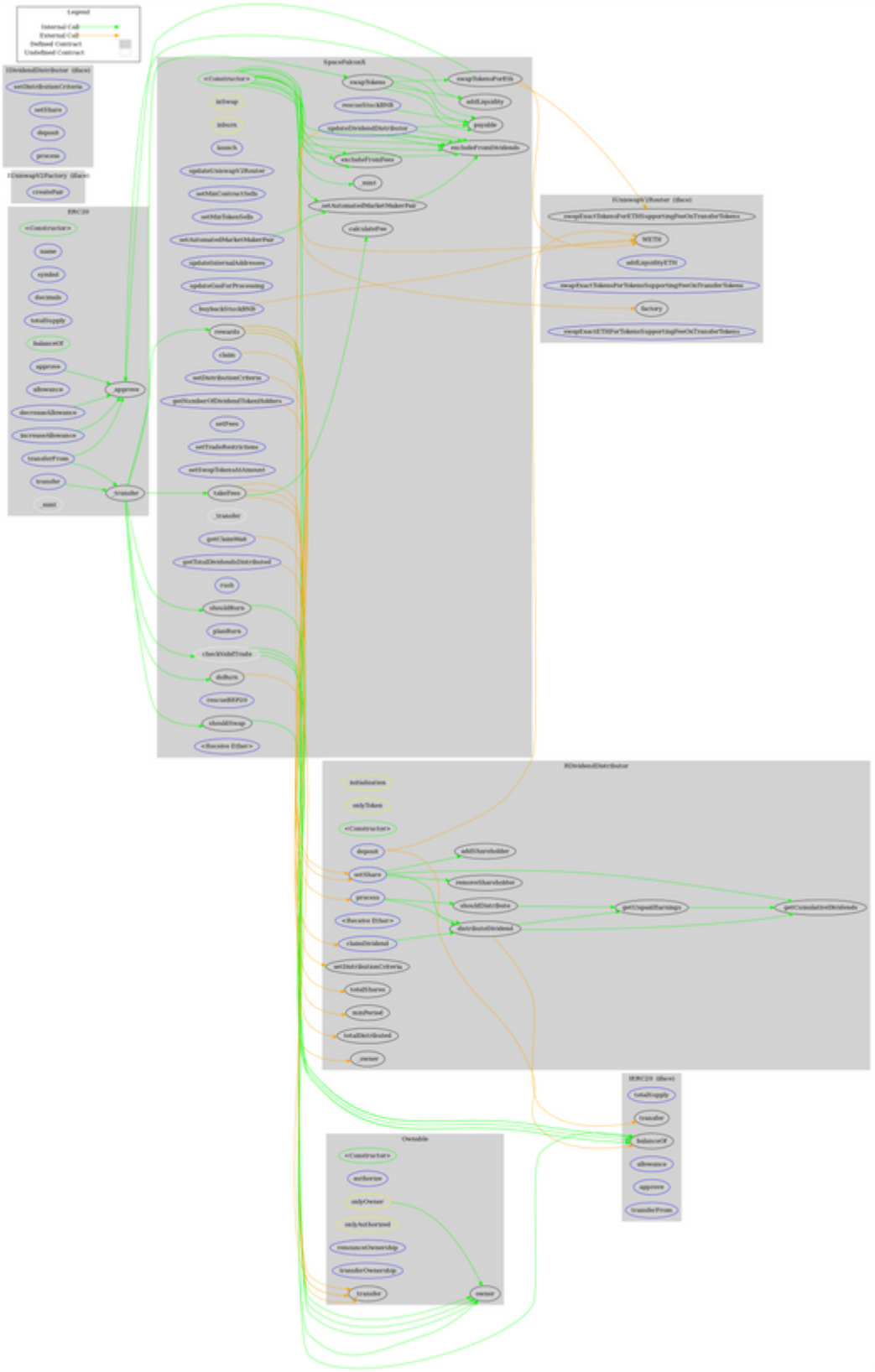
Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IUniswapV2Router	Interface			
	factory	External		-
	WETH	External		-
	addLiquidityETH	External	Payable	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
IUniswapV2Factory	Interface			
	createPair	External	✓	-
IDividendDistributor	Interface			
	setDistributionCriteria	External	✓	-
	setShare	External	✓	-
	deposit	External	Payable	-
	process	External	✓	-

Ownable	Implementation			
	<Constructor>	Public	✓	-
	owner	Public		-
	authorize	External	✓	onlyOwner
	renounceOwnership	External	✓	onlyOwner
	transferOwnership	External	✓	onlyOwner
ERC20	Implementation	IERC20		
	<Constructor>	Public	✓	-
	name	External		-
	symbol	External		-
	decimals	External		-
	totalSupply	External		-
	balanceOf	Public		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
	increaseAllowance	External	✓	-
	decreaseAllowance	External	✓	-
	_transfer	Internal	✓	
	_mint	Internal	✓	
	_approve	Internal	✓	
RDividendDistributor	Implementation	IDividendDistributor		
	<Constructor>	Public	✓	-
	setDistributionCriteria	External	✓	onlyToken
	setShare	External	✓	onlyToken
	deposit	External	Payable	-
	<Receive Ether>	External	Payable	-
	process	External	✓	-
	shouldDistribute	Internal		
	distributeDividend	Internal	✓	
	claimDividend	External	✓	-

	getUnpaidEarnings	Public		-
	getCumulativeDividends	Internal		
	addShareholder	Internal	✓	
	removeShareholder	Internal	✓	
SpaceFalconX	Implementation	ERC20, Ownable		
	<Constructor>	Public	✓	ERC20
	launch	External	✓	onlyOwner
	updateDividendDistributor	External	✓	onlyOwner
	updateUniswapV2Router	External	✓	onlyOwner
	setMinContractSells	External	✓	onlyAuthorized
	setMinTokenSells	External	✓	onlyAuthorized
	excludeFromFees	Public	✓	onlyOwner
	excludeFromDividends	Public	✓	onlyOwner
	setAutomatedMarketMakerPair	External	✓	onlyOwner
	_setAutomatedMarketMakerPair	Internal	✓	
	updateInternalAddresses	External	✓	onlyAuthorized
	updateGasForProcessing	External	✓	onlyOwner
	setDistributionCriteria	External	✓	onlyOwner
	getClaimWait	External		-
	getTotalDividendsDistributed	External		-
	claim	External	✓	-
	getNumberOfDividendTokenHolders	External		-
	setFees	External	✓	onlyOwner
	setTradeRestrictions	External	✓	onlyOwner
	setSwapTokensAtAmount	External	✓	onlyOwner
	checkValidTrade	Internal	✓	
	_transfer	Internal	✓	
	rewards	Internal	✓	
	takeFees	Internal	✓	
	rush	External	✓	onlyAuthorized
	calculateFee	Internal	✓	
	shouldBurn	Internal		
	planBurn	External	✓	onlyAuthorized

	doBurn	Internal	✓	inburn
	shouldSwap	Internal		
	rescueStuckBNB	External	✓	onlyAuthorized
	rescueBEP20	External	✓	onlyAuthorized
	swapTokens	Internal	✓	inSwap
	swapTokensForEth	Internal	✓	
	addLiquidity	Internal	✓	
	buybackStuckBNB	External	✓	onlyAuthorized
	<Receive Ether>	External	Payable	-

Contract Flow



Domain Info

Domain Name	spacefalconx.com
Registry Domain ID	2701686615_DOMAIN_COM-VRSN
Creation Date	2022-06-06T02:52:56Z
Updated Date	2022-06-06T02:52:58Z
Registry Expiry Date	2023-06-06T02:52:56Z
Registrar WHOIS Server	whois.hostinger.com
Registrar URL	https://www.hostinger.com
Registrar	Hostinger, UAB
Registrar IANA ID	1636

The domain has been created about 9 hours 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

There are some functions that can be abused by the owner like misusing the contract configuration and manipulating fees. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats.

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