Audit Report

SafeSea

January 2022

Type	BEP20
Network	BSC
Address	0x55D6a39c4bB3211b02a860Cc53aa205dbA132c79
Audited by	© coinscope

Table of Contents

Contract Review

Contract Name	SafeSea
Compiler Version	v0.8.11+commit.d7f03943
Optimization	800 runs
Licence	
Explorer	https://bscscan.com/token/0x55D6a39c4bB3211b02a860Cc53aa205dbA132c79
Symbol	SAFESEA
Decimals	9
Total Supply	1,000,000,000,000,000
Source	contracts/SafeSea.sol, @pancakeswap/v2-core/contracts/interfaces/IPancakeFactory.sol, @pancakeswap/v2-core/contracts/interfaces/IPancakePair.sol, @pancakeswap/v2-periphery/contracts/interfaces/IPancakeRouter01.sol, @pancakeswap/v2-periphery/contracts/interfaces/IPancakeRouter02.sol, @openzeppelin/contracts/token/ERC20/IERC20.sol, @openzeppelin/contracts/access/Ownable.sol, @openzeppelin/contracts/utils/Address.sol, @openzeppelin/contracts/utils/Context.sol
Domain	

Audit Updates

Initial Audit	14th January 2022
Corrected	19th January 2022

Contract Analysis

● Critical ● Medium ● Minor ● Pass

Severity	Code	Description	Status
●	ST	Contract Owner is not able to stop or pause transactions	Semi resolved
●	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%)	Semi resolved
●	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	

ST - Stop Transactions

Criticality	medium
Location	SafeSea.sol#L304
Status	Semi resolved

Description

The contract owner has the authority to stop transactions for all users excluding the owner. The owner may take advantage of it by setting the `_maxTxAmount` to zero.

```
if (from != owner() && to != owner()) require(amount <= _maxTxAmount, "Transfer amount exceeds the maxTxAmount.");
```

Recommendation

The contract could embody a check for not allowing setting the _maxTxAmount less than a reasonable amount. A suggested implementation could check that the maximum amount should be more than a fixed percentage of the total supply.

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.

Team Update 19 January 2022

Considering the critical security concern about privileged ownership, the contract does not have an update function. Thus, it will be impossible to update directly. Renouncing ownership of the contract will result in an inability to include exchanges. Our plan is to create a periphery multisig contract for the contract owner functions and assign the contract owner to it.

As of now, there will be no transfer of ownership. However, team members will be evaluated through KYC by the IdoPresales entity, and their result will be published on both their website and SafeSea's website. The multisig is underway and will be completed as soon as possible, adding it as an act of good faith.

We have taken serious steps towards further risk mitigation by initially starting this project with a fair launch hosted on DxSale, where the LP being immediately locked out of the gate. SafeSea is very different from other projects, and our differences provide more security for the community vs. anonymous teams and projects. Risks regarding “rug-pulls” or anything else is mitigated, since every member of SafeSea would be subject to litigation and likely a swift prison sentence.

ELFM - Exceed Limit Fees Manipulation

Criticality	critical
Location	SafeSea.sol#L195,L199
Status	Semi resolved

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 `setTaxFeePercent` function with a high percentage value.

```
 function setTaxFeePercent(uint256 taxFee) external onlyOwner() {
    _taxFee = taxFee;
}
```

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.

Team Update 19 January 2022

We are aware of what this action may entail. However, we have plans to develop a protocol upgrade, in which token migration will be indispensable. Therefore, we believe that this feature is essential to encourage our holders to migrate their tokens to the second version of the contract, applying a higher fee on the first one to stop using it.

All actions will be conducted with prior notice to the community. In addition, as soon as the project is launched, and we have enough capital, we will become a

legally registered entity in accordance with the law and jurisdictions in which we operate. This should be considered when looking at the SafeSea project.

We have publicly expressed our goals and intentions of why we will retain custody of the contract. The functions allow additional control for the SafeSea team to make continued strategic plays regarding the long-term growth of the community and the project.

Contract Diagnostics

Critical Medium Minor

Severity	Code	Description	Status
	L01	Public Function could be Declared External	Resolved
	L04	Conformance to Solidity Naming Conventions	Resolved
	L09	Dead Code Elimination	Resolved
	L07	Missing Events Arithmetic	Resolved

L01 - Public Function could be Declared External

Criticality	minor
Location	@openzeppelin/contracts/access/Ownable.sol#L288,L207,L191 and 17 more
Status	Resolved

Description

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

```
isExcludedFromFee
setSwapAndLiquidateEnabled
includeInFee
...
```

Recommendation

Use the external attribute for functions never called from the contract

Team Update 19 January 2022

The team acknowledged the finding, and given the deployed contract cannot be updated, decided to retain the code base unchanged.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	@pancakeswap/v2-core/contracts/interfaces/IPancakePair.sol#L51,L45,L44 and 13 more
Status	Resolved

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.

```
numTokensSellToAddToLiquidity
_pancakePair
_pancakeRouter
...
```

Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.8.11/style-guide.html#naming-conventions

Team Update 19 January 2022

The team acknowledged the finding, and given the deployed contract cannot be updated, decided to retain the code base unchanged.

L09 - Dead Code Elimination

Criticality	minor
Location	@openzeppelin/contracts/utils/Address.sol#L137,L21,L196 and 10 more
Status	Resolved

Description

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

```
reflectionFromToken
_msgData
verifyCallResult
...
```

Recommendation

Remove unused functions.

Team Update 19 January 2022

The team acknowledged the finding, and given the deployed contract cannot be updated, decided to retain the code base unchanged.

L07 - Missing Events Arithmetic

Criticality	minor
Location	contracts/SafeSea.sol#L203,L199,L195
Status	Resolved

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.

```
_maxTxAmount = (_tTotal * maxTxPercent) / 10 ** 2
_liquidityFee = liquidityFee
_taxFee = taxFee
```

Recommendation

Emit an event for critical parameter changes.

Team Update 19 January 2022

The team acknowledged the finding, and given the deployed contract cannot be updated, decided to retain the code base unchanged.

Contract Functions

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_transferOwnership	Internal	✓	
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	verifyCallResult	Internal		
Context	Implementation			

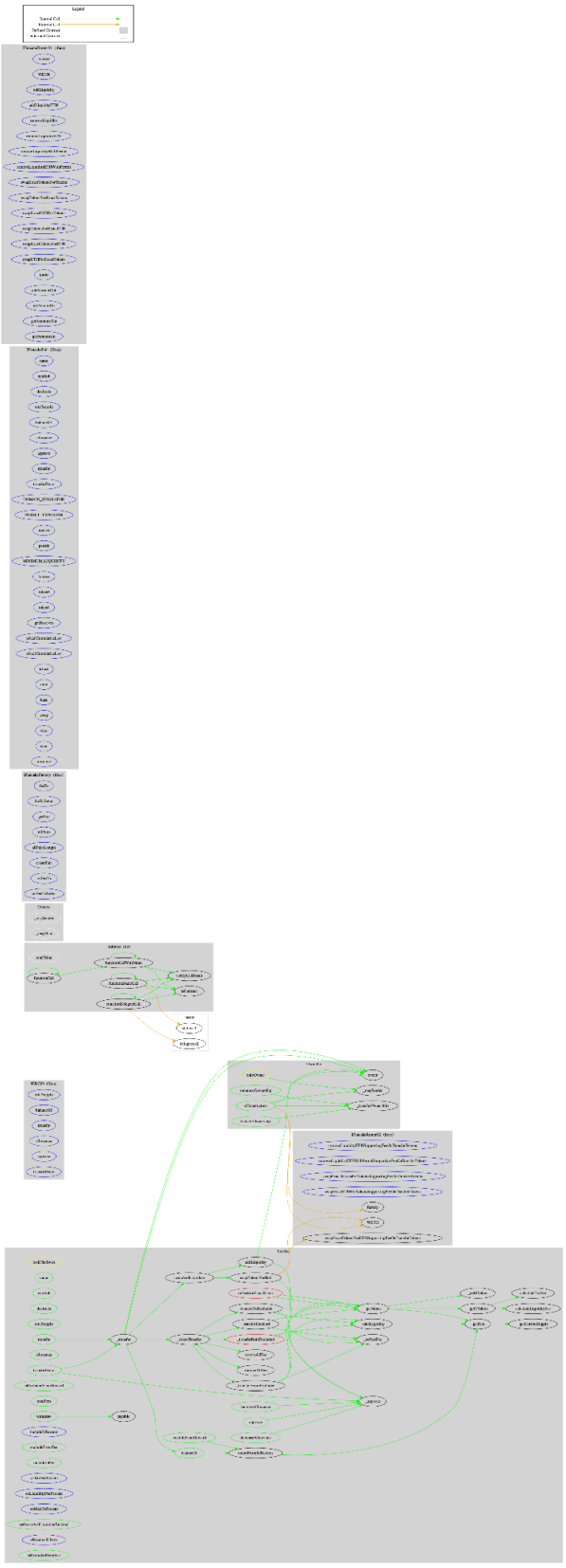
	_msgSender	Internal		
	_msgData	Internal		
IPancakeFactory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
IPancakePair	Interface			
	name	External		-
	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	✓	-
IPancakeRouter01	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		-
IPancakeRouter02	Interface	IPancakeRouter01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-

	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
SafeSea	Implementation	IERC20, Ownable		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	isExcludedFromReward	Public		onlyOwner
	totalFees	Public		-
	withdraw	Public	✓	onlyOwner
	reflectionFromToken	Private		
	tokenFromReflection	Private		
	excludeFromReward	Public	✓	onlyOwner
	includeInReward	External	✓	onlyOwner
	_transferBothExcluded	Private	✓	
	excludeFromFee	Public	✓	onlyOwner
	includeInFee	Public	✓	onlyOwner
	setTaxFeePercent	External	✓	onlyOwner
	setLiquidityFeePercent	External	✓	onlyOwner
	setMaxTxPercent	External	✓	onlyOwner
	setSwapAndLiquidateEnabled	Public	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	_reflectFee	Private	✓	
	_getValues	Private		

	_getTValues	Private		
	_getRValues	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	_takeLiquidity	Private	✓	
	calculateTaxFee	Private		
	calculateLiquidityFee	Private		
	removeAllFee	Private	✓	
	restoreAllFee	Private	✓	
	isExcludedFromFee	Public		onlyOwner
	_approve	Private	✓	
	_transfer	Private	✓	
	swapAndLiquidate	Private	✓	lockTheSwap
	swapTokensForBnb	Private	✓	
	addLiquidity	Private	✓	
	_tokenTransfer	Private	✓	
	_transferStandard	Private	✓	
	_transferToExcluded	Private	✓	
	_transferFromExcluded	Private	✓	

Contract Flow



Summary

The Smart Contract analysis reported two issues. There are some functions that can be abused by the owner, like manipulating fees and indirectly stopping the transactions. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats.

Update 19 January 2022

The token team has investigated and replied to all of the findings.

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.

Coinscope 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 Coinscope 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 Coinscope 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 Coinscope team disclaims any liability for the resulting losses.

About Coinscope

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.

Coinscope is aiming to make crypto discoverable and efficient globally. It provides all the essential tools to assist users draw their own conclusions.

The Coinscope.co team

https://www.coinscope.co