

Smart Contract Security Audit V1

ToastFu Smart Contract Audit

Mar 31, 2024



<https://saferico.com/>

business@saferico.com

https://t.me/SFI_ANN

—

Table of Contents

Table of Contents

Background

Project Information

Token Smart Contract Information

Executive Summary

File and Function Level Report

File in Scope:

Issues Checking Status

SWC Attack Analysis

Severity Definitions

Audit Findings

Automatic testing

Testing proves

Inheritance graph

Call graph

Source lines

Risk level

Source units in scope

Capabilities

Unified Modeling Language (UML)

Functions signature

Automatic general report

Conclusion

Disclaimer

Background

The purpose of the audit was to achieve the following:

- Ensure that the smart contract functions as intended.
- Identify potential security issues with the smart contract.

The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

Project Information

- **Platform:** Ethereum
- **Name:** ToastFu
- **Language :** solidity
- **Contract Address:** 0x6668e12A659A643bD5D7561630E17D5433375b2f
- **Code Source:** <https://etherscan.io/token/0x6668e12A659A643bD5D7561630E17D5433375b2f#code>
- **Website:** <https://www.toastfu.com/>
- **Telegram:** <https://t.me/ToastFuPortal>
- **X:** <https://twitter.com/ToastFuTech>
- **Youtube:** <https://www.youtube.com/@ToastFuTech>
- **WhitePaper:** <https://toastfu.gitbook.io/toastfu>

Executive Summary

According to our assessment, the customer`s solidity smart contract is **Well-Secured**.

Well Secured	✓
Secured	
Poor Secured	
Insecure	

Automated checks are with remix IDE. All issues were performed by the team, which included the analysis of code functionality, manual audit found during automated analysis were manually reviewed and applicable vulnerabilities are presented in the audit overview section. The general overview is presented in the Project Information section and all issues found are located in the audit overview section.

Team found 0 critical, 0 high, 0 medium, 2 low, 0 very low-level issues and 0 note in all solidity files of the contract

The files:

ToastFu.sol

Audit Score:

100% secure



File and Function Level Report

File in Scope:

Contract Name	SHA 256 hash	Contract Address
ToastFu.sol	60a0f139635453eb4352dff2fb53bc2877202fc2	0x6668e12A659A643bD5D7561630E17D5433375b2f

- Contract: ToastFu
- Inherit: Context, IERC20, Ownable
- Observation: All passed including security check
- Test Report: **passed**
- Score: **passed**
- Conclusion: **passed**

Function	Test Result	Type / Return Type	Score
allowance	✓	Read / public	Passed
buyFee	✓	Read / public	Passed
decimals	✓	Read / public	Passed
balanceOf	✓	Read / public	Passed
deadWallet	✓	Read / public	Passed
inSwapAndLiquify	✓	Read / public	Passed
maxWallet	✓	Read / public	Passed
minimumTokensBefore Swap	✓	Read / public	Passed
sellFee	✓	Read / public	Passed
name	✓	Read / public	Passed
owner	✓	Read / public	Passed
swapAndLiquifyEnabled	✓	Read / public	Passed
taxWallet	✓	Read / public	Passed

taxTokensCollected	✓	Read / public	Passed
swapOutput	✓	Read / public	Passed
symbol	✓	Read / public	Passed
totalSupply	✓	Read / public	Passed
tradingActive	✓	Read / public	Passed
totalTaxTokensCollecte d	✓	Read / public	Passed
uniswapV2Pair	✓	Read / public	Passed
uniswapV2Router	✓	Read / public	Passed
isExcludedFromFee	✓	Read / public	Passed
approve	✓	Write / public	Passed
transfer	✓	Write / public	Passed
transferFrom	✓	Write / public	Passed
excludedFromFee	✓	Write / public	Passed
includeInFee	✓	Write / public	Passed
decreaseAllowance	✓	Write / public	Passed
increaseAllowance	✓	Write / public	Passed
openTrading	✓	Write / public	Passed
transferOwnership	✓	Write / public	Passed
renounceOwnership	✓	Write / public	Passed
recoverETHfromContrac t	✓	Write / public	Passed
recoverTokensFromCont ract	✓	Write / public	Passed
setSwapAndLiquifyEna bled	✓	Write / public	Passed
setTaxWallet	✓	Write / public	Passed
setTokensToSwap	✓	Write / public	Passed
swapAndLiquify	✓	Write / public	Passed
updateFee	✓	Write / public	Passed

Issues Checking Status

SWC Attack Analysis

The Smart Contract Weakness Classification Registry (SWC Registry) is an implementation of the weakness classification scheme proposed in EIP-1470. It is loosely aligned to the terminologies and structure used in the Common Weakness Enumeration (CWE) for more info check

<https://swcregistry.io/>

No.	Issue Description	Checking Status
136	Unencrypted Private Data On-Chain	Passed
135	Code With No Effects	Passed
134	Message call with hardcoded gas amount	Passed
133	Hash Collisions With Multiple Variable Length Arguments	Passed
132	Unexpected Ether balance	Passed
131	Presence of unused variables	Passed
130	Right-To-Left-Override control character (U+202E)	Passed
129	Typographical Error	Passed
128	DoS with block gas limit.	Passed
127	Arbitrary Jump with Function Type Variable	Passed
126	Insufficient Gas Griefing	Passed
125	Incorrect Inheritance Order	Passed
124	Write to Arbitrary Storage Location	Passed
123	Requirement Violation	Passed
122	Lack of Proper Signature Verification	Passed
121	Missing Protection against Signature Replay Attacks	Passed
120	Weak Sources of Randomness from Chain Attributes	Passed
119	Shadowing State Variables	Passed

118	Incorrect Constructor Name	Passed
117	Signature Malleability	Passed
116	Block values as a proxy for time	Not Passed
115	Authorization through tx.origin	Passed
114	Transaction Order Dependence	Passed
113	DoS with Failed Call	Passed
112	Delegatecall to Untrusted Callee	Passed
111	Use of Deprecated Solidity Functions	Passed
110	Assert Violation	Passed
109	Uninitialized Storage Pointer	Passed
108	State Variable Default Visibility	Passed
107	Reentrancy	Passed
106	Unprotected SELFDESTRUCT Instruction	Passed
105	Unprotected Ether Withdrawal	Passed
104	Unchecked Call Return Value	Passed
103	Floating Pragma	Passed
102	Outdated Compiler Version	Passed
101	Integer Overflow and Underflow	Passed
100	Function Default Visibility	Passed

Severity Definitions

Risk Level	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to tokens loss etc.
High	High-level vulnerabilities are difficult to exploit; however, they also have significant impact on smart contract execution, e.g. public access to crucial functions
Medium	Medium-level vulnerabilities are important to fix; however, they can't lead to tokens lose
Low	Low-level vulnerabilities are mostly related to outdated, unused etc. code snippets, that can't have significant impact on execution
Note	Lowest-level vulnerabilities, code style violations and info statements can't affect smart contract execution and can be ignored.

Audit Findings

Critical:

No Critical severity vulnerabilities were found.

High:

No High severity vulnerabilities were found.

Medium:

No Medium severity vulnerabilities were found.

Low:

Use of block.timestamp for comparisons

The value of block.timestamp can be manipulated by the miner. And conditions with strict equality is difficult to achieve - block.timestamp.

```
function swapTokensForEth(uint256 tokenAmount) private {
    // generate the uniswap pair path of token -> weth
    address[] memory path = new address[](2);
    path[0] = address(this);
    path[1] = WETH;
    _approve(address(this), address(uniswapV2Router),
tokenAmount);

    // make the swap

    uniswapV2Router.swapExactTokensForETHSupportingFeeOnTransferTokens(
        tokenAmount,
        0, // accept any amount of ETH
        path,
        address(this), // The contract
        block.timestamp
    );

    emit SwapTokensForETH(tokenAmount, path);
}
```

Recommendation

Avoid use of block.timestamp.

Status: **Acknowledged.**

#Owner privileges (In the period when the owner isn't renounced)

Description

The owner can change the buy and sell Fees max to 5%.

The owner can open the trading.

The owner can include / exclude any address in/ from Fees.

```
function excludeFromFee(address account) external onlyOwner {
    _isExcludedFromFee[account] = true;
    emit AuditLog(
        "We have excluded the following walled in fees:",
        account
    ); }

function includeInFee(address account) external onlyOwner {
    _isExcludedFromFee[account] = false;
    emit AuditLog(
        "We have including the following walled in fees:",
        account
    ); }

function updateFees(uint256 _buyFee, uint256 _sellFee) external onlyOwner {
    require(_buyFee <= 5, "Must keep buy fees at 5% or less");
    require(_sellFee <= 5, "Must keep sell fees at 5% or less");
    buyFee = _buyFee;
    sellFee = _sellFee;}

function openTrading() external onlyOwner {
    tradingActive = true;}
```

Remediation

Make these functions internal in next version or the team should announce the investors before doing anything to give them time if they want to do anything.

P.S: This issue is common to the majority of those smart contracts.

Status: **Acknowledged**.

Very Low:

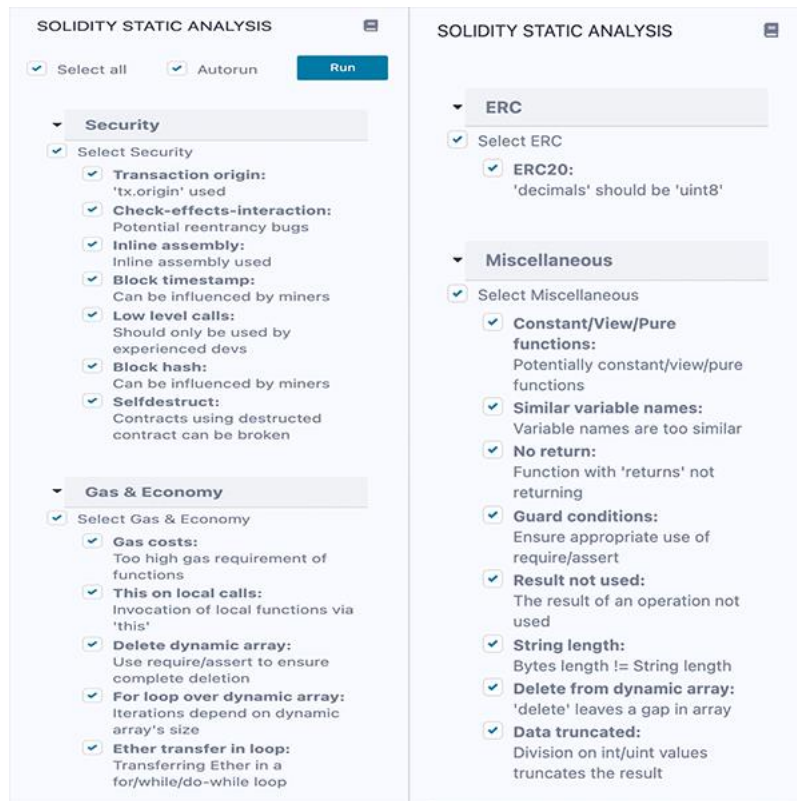
No Very Low severity vulnerabilities were found.

Notes:

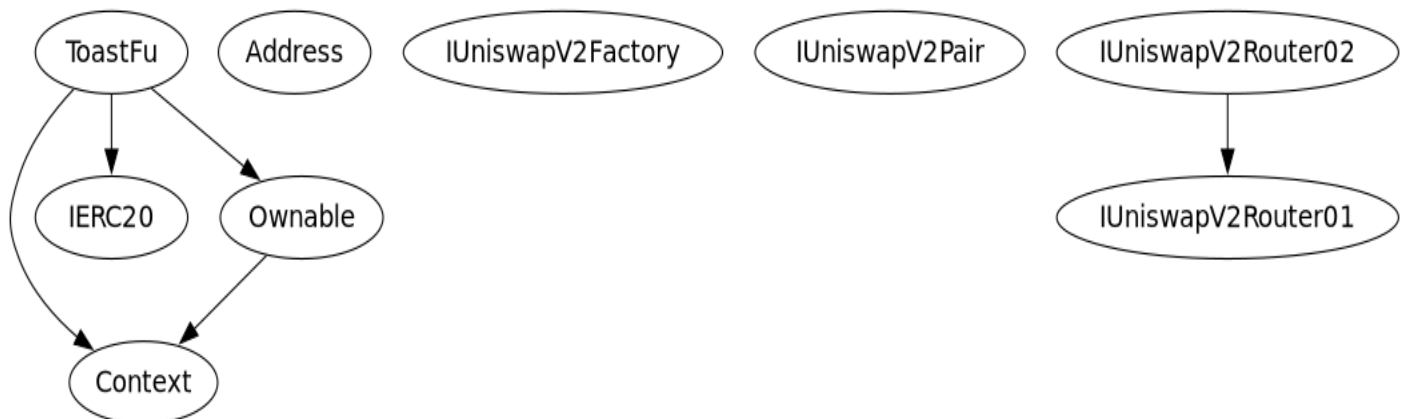
No Notes were found.

Automatic Testing

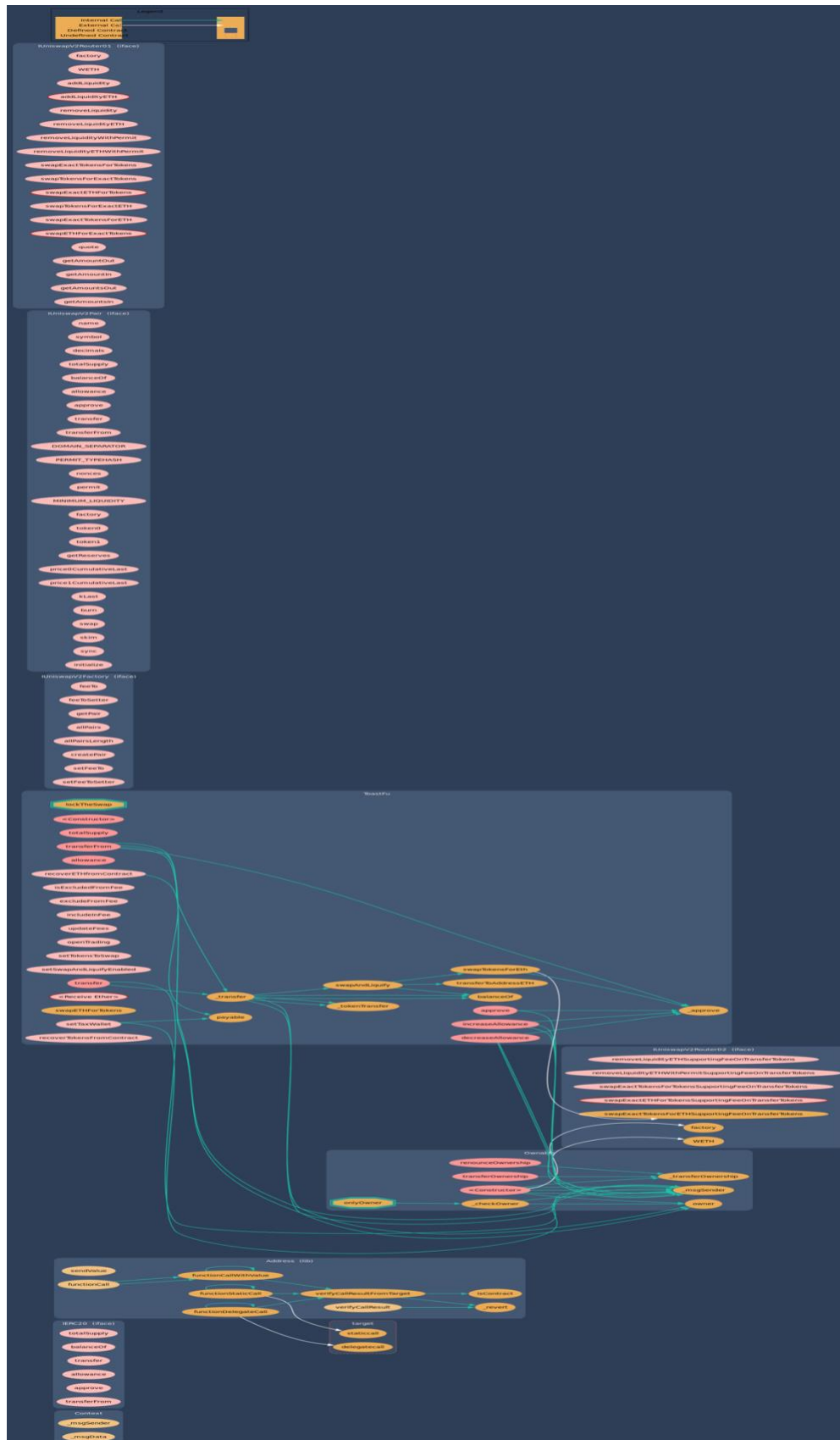
1- SOLIDITY STATIC ANALYSIS



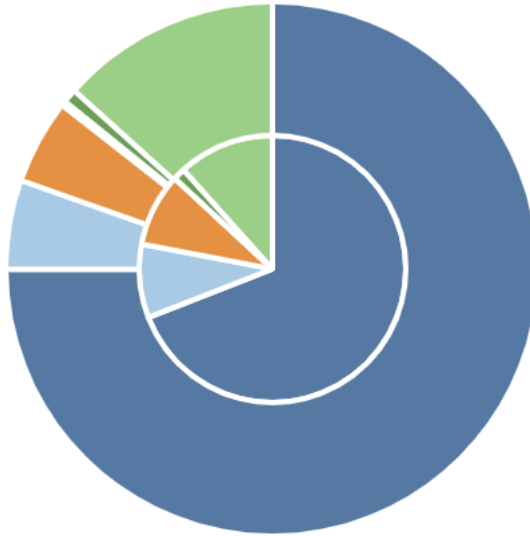
2- Inheritance graph



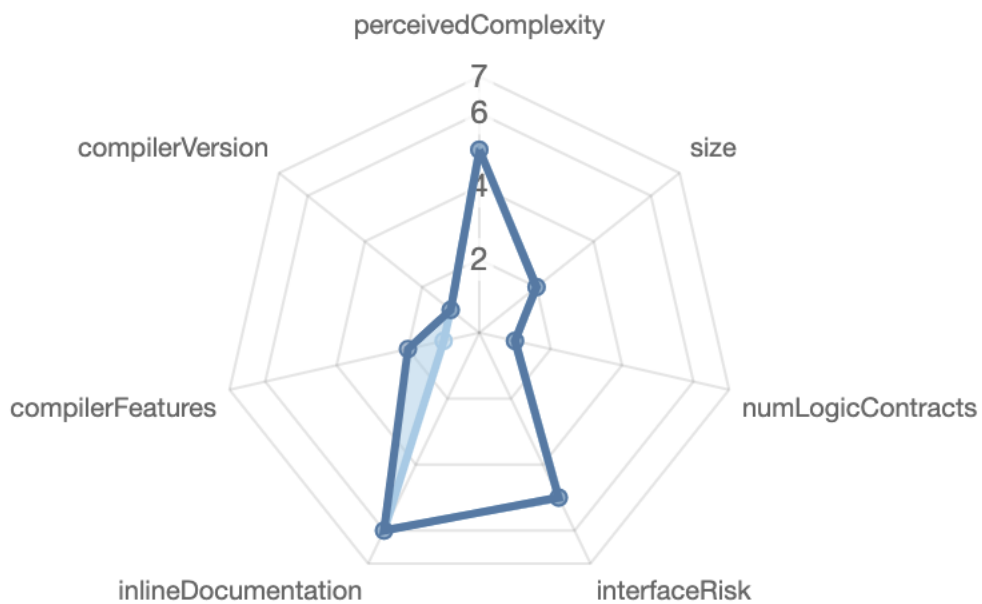
3- Call graph



Source lines







Risk level



Source units in scope

Source Units in Scope

Source Units Analyzed: 1
Source Units in Scope: 1 (100%)

Type	File	Logic Contracts	Interfaces	Lines	nLines	nSLOC	Comment Lines	Complex. Score	Capabilities
	TaxTokenSample.sol	4	5	995	545	427	58	425	
	Totals	4	5	995	545	427	58	425	

Legend: [-]

- **Lines**: total lines of the source unit
- **nLines**: normalized lines of the source unit (e.g. normalizes functions spanning multiple lines)
- **nSLOC**: normalized source lines of code (only source-code lines; no comments, no blank lines)
- **Comment Lines**: lines containing single or block comments
- **Complexity Score**: a custom complexity score derived from code statements that are known to introduce code complexity (branches, loops, calls, external interfaces, ...)

Capabilities

Components

 Contracts	 Libraries	 Interfaces	 Abstract
1	1	5	2

Exposed Functions

This section lists functions that are explicitly declared public or payable. Please note that getter methods for public stateVars are not included.

 Public	 Payable
87	5

External	Internal	Private	Pure	View
75	77	7	12	34

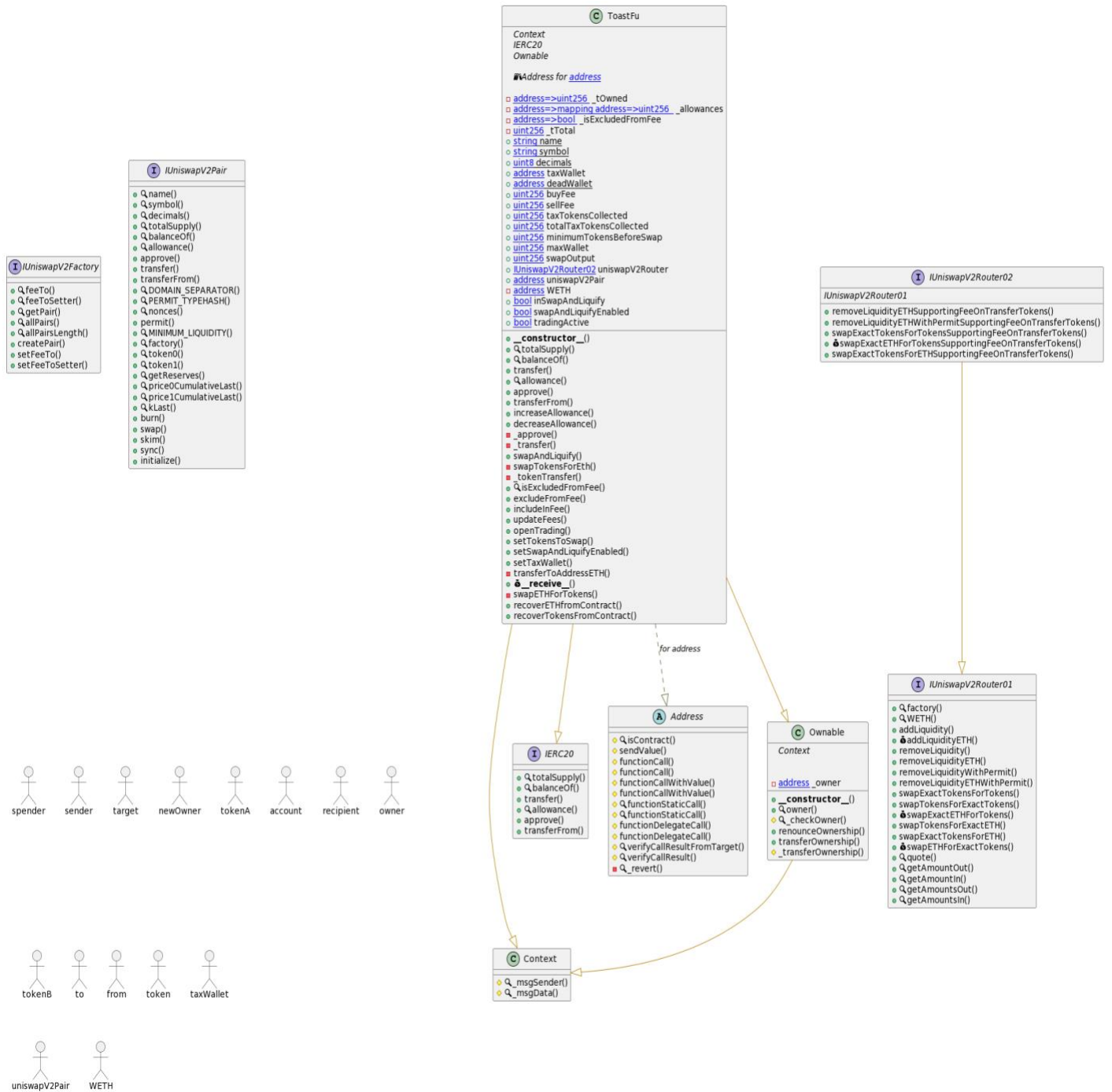
StateVariables

Total	 Public
23	17

Capabilities

Solidity Versions observed	 Experimental Features	 Can Receive Funds	 Uses Assembly	 Has Destroyable Contracts	
<div>0.8.19</div>		<div>yes</div>	<div>yes</div> <div>(1 asm blocks)</div>		
 Transfers ETH	 Low-Level Calls	 DelegateCall	 Uses Hash Functions	 ECREcover	 New/Create/Create2
<div>yes</div>		<div>yes</div>			

Unified Modeling Language (UML)



Functions signature

Function Name	Sighash	Function Signature
totalSupply	18160ddd	totalSupply()
balanceOf	70a08231	balanceOf(address)
transfer	a9059cbb	transfer(address,uint256)
allowance	dd62ed3e	allowance(address,address)
approve	095ea7b3	approve(address,uint256)
transferFrom	23b872dd	transferFrom(address,address,uint256)
owner	8da5cb5b	owner()
renounceOwnership	715018a6	renounceOwnership()
transferOwnership	f2fde38b	transferOwnership(address)
feeTo	017e7e58	feeTo()
feeToSetter	094b7415	feeToSetter()
getPair	e6a43905	getPair(address,address)
allPairs	1e3dd18b	allPairs(uint256)
allPairsLength	574f2ba3	allPairsLength()
createPair	c9c65396	createPair(address,address)
setFeeTo	f46901ed	setFeeTo(address)
setFeeToSetter	a2e74af6	setFeeToSetter(address)
name	06fdde03	name()
symbol	95d89b41	symbol()
decimals	313ce567	decimals()
totalSupply	18160ddd	totalSupply()
balanceOf	70a08231	balanceOf(address)
allowance	dd62ed3e	allowance(address,address)
approve	095ea7b3	approve(address,uint256)
transfer	a9059cbb	transfer(address,uint256)
transferFrom	23b872dd	transferFrom(address,address,uint256)
DOMAIN_SEPARATOR	3644e515	DOMAIN_SEPARATOR()
PERMIT_TYPEHASH	30adf81f	PERMIT_TYPEHASH()
nonces	7ecebe00	nonces(address)
permit	d505accf	permit(address,address,uint256,uint256,uint8,bytes32,bytes32)
MINIMUM_LIQUIDITY	ba9a7a56	MINIMUM_LIQUIDITY()
factory	c45a0155	factory()
token0	0dfe1681	token0()
token1	d21220a7	token1()
getReserves	0902f1ac	getReserves()
price0CumulativeLast	5909c0d5	price0CumulativeLast()
price1CumulativeLast	5a3d5493	price1CumulativeLast()
kLast	7464fc3d	kLast()
burn	89afcb44	burn(address)
swap	022c0d9f	swap(uint256,uint256,address,bytes)
skim	bc25cf77	skim(address)
sync	fff6cae9	sync()
initialize	c4d66de8	initialize(address)
factory	c45a0155	factory()
WETH	ad5c4648	WETH()
addLiquidity	e8e33700	addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256)
addLiquidityETH	f305d719	addLiquidityETH(address,uint256,uint256,uint256,address,uint256)
removeLiquidity	baa2abde	removeLiquidity(address,address,uint256,uint256,uint256,address,uint256)
removeLiquidityETH	02751cec	removeLiquidityETH(address,uint256,uint256,uint256,address,uint256)

```

| removeLiquidityWithPermit | 2195995c |
removeLiquidityWithPermit(address,address,uint256,uint256,uint256,address,uint256,boo
ool,uint8,bytes32,bytes32) |
| removeLiquidityETHWithPermit | ded9382a |
removeLiquidityETHWithPermit(address,uint256,uint256,uint256,address,uint256,bool,u
int8,bytes32,bytes32) |
| swapExactTokensForTokens | 38ed1739 |
swapExactTokensForTokens(uint256,uint256,address[],address,uint256) |
| swapTokensForExactTokens | 8803dbee |
swapTokensForExactTokens(uint256,uint256,address[],address,uint256) |
| swapExactETHForTokens | 7ff36ab5 |
swapExactETHForTokens(uint256,address[],address,uint256) |
| swapTokensForExactETH | 4a25d94a |
swapTokensForExactETH(uint256,uint256,address[],address,uint256) |
| swapExactTokensForETH | 18cbafe5 |
swapExactTokensForETH(uint256,uint256,address[],address,uint256) |
| swapETHForExactTokens | fb3bdb41 |
swapETHForExactTokens(uint256,address[],address,uint256) |
| quote | ad615dec | quote(uint256,uint256,uint256) |
| getAmountOut | 054d50d4 | getAmountOut(uint256,uint256,uint256) |
| getAmountIn | 85f8c259 | getAmountIn(uint256,uint256,uint256) |
| getAmountsOut | d06ca61f | getAmountsOut(uint256,address[]) |
| getAmountsIn | 1f00ca74 | getAmountsIn(uint256,address[]) |
| removeLiquidityETHSupportingFeeOnTransferTokens | af2979eb |
removeLiquidityETHSupportingFeeOnTransferTokens(address,uint256,uint256,uint256,add
ress,uint256) |
| removeLiquidityETHWithPermitSupportingFeeOnTransferTokens | 5b0d5984 |
removeLiquidityETHWithPermitSupportingFeeOnTransferTokens(address,uint256,uint256,u
int256,address,uint256,bool,uint8,bytes32,bytes32) |
| swapExactTokensForTokensSupportingFeeOnTransferTokens | 5c11d795 |
swapExactTokensForTokensSupportingFeeOnTransferTokens(uint256,uint256,address[],add
ress,uint256) |
| swapExactETHForTokensSupportingFeeOnTransferTokens | b6f9de95 |
swapExactETHForTokensSupportingFeeOnTransferTokens(uint256,address[],address,uint25
6) |
| swapExactTokensForETHSupportingFeeOnTransferTokens | 791ac947 |
swapExactTokensForETHSupportingFeeOnTransferTokens(uint256,uint256,address[],addres
s,uint256) |
| totalSupply | 18160ddd | totalSupply() |
| balanceOf | 70a08231 | balanceOf(address) |
| transfer | a9059cbb | transfer(address,uint256) |
| allowance | dd62ed3e | allowance(address,address) |
| approve | 095ea7b3 | approve(address,uint256) |
| transferFrom | 23b872dd | transferFrom(address,address,uint256) |
| increaseAllowance | 39509351 | increaseAllowance(address,uint256) |
| decreaseAllowance | a457c2d7 | decreaseAllowance(address,uint256) |
| swapAndLiquify | b29ad50a | swapAndLiquify() |
| isExcludedFromFee | 5342acb4 | isExcludedFromFee(address) |
| excludeFromFee | 437823ec | excludeFromFee(address) |
| includeInFee | ea2f0b37 | includeInFee(address) |
| updateFees | 6db79437 | updateFees(uint256,uint256) |
| openTrading | c9567bf9 | openTrading() |
| setTokensToSwap | 461d9476 | setTokensToSwap(uint256) |
| setSwapAndLiquifyEnabled | c49b9a80 | setSwapAndLiquifyEnabled(bool) |
| setTaxWallet | ea414b28 | setTaxWallet(address) |
| recoverETHfromContract | ce831ed5 | recoverETHfromContract() |
| recoverTokensFromContract | e6be4a72 | recoverTokensFromContract(address,uint256)
|

```

Automatic general report

Files Description Table

File Name	SHA-1 Hash
/Users/macbook/Desktop/smart contracts/ToastFu.sol	60a0f139635453eb4352dff2fb53bc2877202fc2

Contracts Description Table






Contract	Type	Bases		
:-----: :-----: :-----: :-----: :-----:				
L	**Function Name**	**Visibility**	**Mutability**	
Modifiers				
Context	Implementation			
L	_msgSender	Internal		
L	_msgData	Internal		
IERC20	Interface			
L	totalSupply	External		NO
L	balanceOf	External		NO
L	transfer	External		NO
L	allowance	External		NO
L	approve	External		NO
L	transferFrom	External		NO
Address	Library			
L	isContract	Internal		
L	sendValue	Internal		
L	functionCall	Internal		
L	functionCall	Internal		
L	functionCallWithValue	Internal		
L	functionCallWithValue	Internal		
L	functionStaticCall	Internal		
L	functionStaticCall	Internal		
L	functionDelegateCall	Internal		
L	functionDelegateCall	Internal		
L	verifyCallResultFromTarget	Internal		
L	verifyCallResult	Internal		
L	_revert	Private		
Ownable	Implementation	Context		
L	<Constructor>	Public		NO
L	owner	Public		NO
L	_checkOwner	Internal		
L	renounceOwnership	Public		onlyOwner
L	transferOwnership	Public		onlyOwner
L	_transferOwnership	Internal		
IUniswapV2Factory	Interface			
L	feeTo	External		NO
L	feeToSetter	External		NO
L	getPair	External		NO

```

| L | allPairs | External | | NO |
| L | allPairsLength | External | | NO |
| L | createPair | External | | NO |
| L | setFeeTo | External | | NO |
| L | setFeeToSetter | External | | NO |
| | | |
| **IUniswapV2Pair** | Interface | | |
| L | name | External | | NO |
| L | symbol | External | | NO |
| L | decimals | External | | NO |
| L | totalSupply | External | | NO |
| L | balanceOf | External | | NO |
| L | allowance | External | | NO |
| L | approve | External | | NO |
| L | transfer | External | | NO |
| L | transferFrom | External | | NO |
| L | DOMAIN_SEPARATOR | External | | NO |
| L | PERMIT_TYPEHASH | External | | NO |
| L | nonces | External | | NO |
| L | permit | External | | NO |
| L | MINIMUM_LIQUIDITY | External | | NO |
| L | factory | External | | NO |
| L | token0 | External | | NO |
| L | token1 | External | | NO |
| L | getReserves | External | | NO |
| L | price0CumulativeLast | External | | NO |
| L | price1CumulativeLast | External | | NO |
| L | kLast | External | | NO |
| L | burn | External | | NO |
| L | swap | External | | NO |
| L | skim | External | | NO |
| L | sync | External | | NO |
| L | initialize | External | | NO |
| | | |
| **IUniswapV2Router01** | Interface | | |
| L | factory | External | | NO |
| L | WETH | External | | NO |
| L | addLiquidity | External | | NO |
| L | addLiquidityETH | External | | NO |
| L | removeLiquidity | External | | NO |
| L | removeLiquidityETH | External | | NO |
| L | removeLiquidityWithPermit | External | | NO |
| L | removeLiquidityETHWithPermit | External | | NO |
| L | swapExactTokensForTokens | External | | NO |
| L | swapTokensForExactTokens | External | | NO |
| L | swapExactETHForTokens | External | | NO |
| L | swapTokensForExactETH | External | | NO |
| L | swapExactTokensForETH | External | | NO |
| L | swapETHForExactTokens | External | | NO |
| L | quote | External | | NO |
| L | getAmountOut | External | | NO |
| L | getAmountIn | External | | NO |
| L | getAmountsOut | External | | NO |
| L | getAmountsIn | External | | NO |
| | | |
| **IUniswapV2Router02** | Interface | IUniswapV2Router01 | | |
| L | removeLiquidityETHSupportingFeeOnTransferTokens | External | | NO |
| L | removeLiquidityETHWithPermitSupportingFeeOnTransferTokens | External | | NO



```

```

| NO! |
| L | swapExactTokensForTokensSupportingFeeOnTransferTokens | External ! |  | NO! |
|
| L | swapExactETHForTokensSupportingFeeOnTransferTokens | External ! |  | NO! |
| L | swapExactTokensForETHSupportingFeeOnTransferTokens | External ! |  | NO! |
|
|
| **TaxToken** | Implementation | Context, IERC20, Ownable |||
| L | <Constructor> | Public ! |  | NO! |
| L | totalSupply | Public ! | | NO! |
| L | balanceOf | Public ! | | NO! |
| L | transfer | Public ! |  | NO! |
| L | allowance | Public ! | | NO! |
| L | approve | Public ! |  | NO! |
| L | transferFrom | Public ! |  | NO! |
| L | increaseAllowance | Public ! |  | NO! |
| L | decreaseAllowance | Public ! |  | NO! |
| L | _approve | Private  |  |
| L | _transfer | Private  |  |
| L | swapAndLiquify | Public ! |  | lockTheSwap |
| L | swapTokensForEth | Private  |  |
| L | _tokenTransfer | Private  |  |
| L | isExcludedFromFee | External ! | | NO! |
| L | excludeFromFee | External ! |  | onlyOwner |
| L | includeInFee | External ! |  | onlyOwner |
| L | updateFees | External ! |  | onlyOwner |
| L | openTrading | External ! |  | onlyOwner |
| L | setTokensToSwap | External ! |  | onlyOwner |
| L | setSwapAndLiquifyEnabled | External ! |  | onlyOwner |
| L | setTaxWallet | External ! |  | NO! |
| L | transferToAddressETH | Private  |  |
| L | <Receive Ether> | External ! |  | NO! |
| L | swapETHForTokens | Private  |  |
| L | recoverETHfromContract | External ! |  | onlyOwner |
| L | recoverTokensFromContract | External ! |  | onlyOwner |

```

Legend

Symbol	Meaning
:-----:	-----
	Function can modify state
	Function is payable

Conclusion

The contracts are written systematically. Team found no critical issues. So, it is good to go for production.

Since possible test cases can be unlimited and developer level documentation (code flow diagram with function level description) not provided, for such an extensive smart contract protocol, we provide no such guarantee of future outcomes. We have used all the latest static tools and manual observations to cover maximum possible test cases to scan Everything.

Security state of the reviewed contract is “Well Secured”.

- ✓ No volatile code.
- ✓ No high severity issues were found.

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as of the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against the team on the basis of what it says or doesn't say, or how team produced it, and it is important for you to conduct your own independent investigations before making any decisions. team go into more detail on this in the below disclaimer below – please make sure to read it in full.

By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice. No one shall have any right to rely on the report or its contents, and Saferico and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (Saferico s) owe no duty of care towards you or any other person, nor does Saferico make any warranty or representation to any person on the accuracy or completeness of the report. The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and Saferico hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, Saferico hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against Saferico, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report. The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.