RugFreeCoins Audit

Thoreum V2 Token
Smart Contract Security Audit
January 17, 2022

Contents

Audit details

Audited project

Thoreum V2 (Thoreum Multi-chain Venture Capital) Token

Contract Address

0xF2A92BC1Cf798fF4De14502a9C6FdA58865e8D5d

Client contact

Thoreum Team

Blockchain

Binance smart chain

Project website

https://thoreum.finance/

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at 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 us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – please make sure to read it in full.

DISCLAIMER: 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 Rugfreecoins and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (Rugfreecoins) owe no duty of care towards you or any other person, nor does Rugfreecoins 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 Rugfreecoins 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, Rugfreecoins hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against Rugfreecoins, 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.

Background

Rugfreecoins was commissioned by Thoreum Token to perform an audit of the smart contract.

https://bscscan.com/token/0xF2A92BC1Cf798fF4De14502a9C6FdA58865e8D5d

The focus of this audit is to verify that the smart contract is secure, resilient and working according to the specifications.

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

About the project

Thoreum V2 is a token built on the Binance Smart Chain that is with an innovative investment use case the main purpose of which is to seek out constant revenue sources and heading towards building an even greater Community with a multichain Venture Capital that rewards its HOLDers by leveraging multi-chain investment opportunities. With impressive tokenomics, in the last 6 months, Thoreum has been accumulating more than $4 Million USD, and with THOREUM v2, the team will maximize profits from this fund by investing in multiple chains and returning the profits to $Thoreum holders. Each transaction, purchase incurs a 13% fee, and sale incurs a 26% fee.   .

Features

❖ The **Thoreum V2 rewards in BUSD** will be distributed among every holder proportional to how many tokens each individual holds in values of **4.5% when buying and 9% when selling**.

❖ The **sustainability fee of 4.5% when buying and 9% when selling for** treasury/buyback is what allows Thoreum V2 to hold the aforementioned promise. Tokens will be swapped into BNB and will be sent to a wallet when the swap point is reached. This fund will be bridged to multi-chains (such as Avalanche, Polygon, Fantom, etc.) to invest in the most profitable opportunities.

❖ The other component of sustainability fee of 1.5% when buying and 3% when the selling will be allocated for marketing, dev, mods, servers, and other costs. This way, Thoreum V2 will have enough funds to promote the coin and spend for future development without selling tokens as the traditional way.

❖ The additional component included under the sustainability section is a **liquidity fee of 1.5% when buying and 3% when selling**, which is a redistribution mechanism that ensures the trading pool always has sufficient liquidity.

❖ Thoreum V2 has a burn strategy that benefits and rewards those who invest long-term. **A 1% fee will be charged when buying and 2% when selling** and sent to the burn wallet. This feature slowly reduces supply making each Thoreum V2 more and more valuable.

Tokenomics

13% fee when buying

- ❖ 4.5% of trade goes to holders' pockets in BUSD.
- ❖ 4.5% of trade goes to the treasury/buyback wallet in BNB.
- ❖ 1.5% of trade goes to the liquidity pool.
- ❖ 1.5% of trade goes to the marketing/dev wallet in BNB.
- ❖ 1% of trade goes to the burn wallet.

26% fee when selling

- ❖ 9% of trade goes to holders' pockets in BUSD.
- ❖ 9% of trade goes to the treasury/buyback wallet in BNB.
- ❖ 3% of trade goes to the liquidity pool.
- ❖ 3% of trade goes to the marketing/dev wallet in BNB.
- ❖ 2% of trade goes to the burn wallet.

10% fee when transferring from wallet to wallet

Roadmap

The detailed upgrade plan from Thoreum-to-Thoreum v2

- ❖ Thoreum v2 contract & audit (in progress)
- ❖ Disable v1 buyback & burn to finalize the circulating supply of Thoreum v2 (done)
  - Total supply of Thoreum v1: 5,000,000,000
  - Total Thoreum v1 burned: 3,156,410,748.848214249779647375 (https://bscscan.com/token/0x580de58c1bd593a43dadcf0a739d504621817c05?a=0x000000000000000000000000000000000000dead)
  - Circulating supply of Thoreum v1: 1,843,588,800
  - Circulating supply of Thoreum v2 will be: 18,435,888 (1/100 of Thoreum v1)
- ❖ Announce Thoreum v2 tokenomics: here
- ❖ Enable Thoreum v1 to Thoreum v2 swap 1:100, Jan 16, 2022
  - Trading of Thoreum v1 will be disabled, token transfer from wallet to wallet will also be disabled. Liquidity of Thoreum-BNB, Thoreum-Cake, Thoreum USDT LP will not be able to break anymore.
  - Swap 1:100 portal will be enabled at Thoreum.finance, users start to exchange Thoreum v1 to Thoreum v2
  - Thor's Thunder 8,300 BNB will be taken out from contract by buyback&burn the Thoreum v1 tokens from (2).
  - v1 Liquidity in BNB will be drained by selling the Thoreum v1 tokens from (2).
  - v2 Liquidity will be created using the BNB from (4). Thoreum v2 price will be x100 as Thoreum v1 price at the time of (1).
- ❖ Enable Thoreum v2 trading, Jan 16 2022.
- ❖ Enable Thoreum v2 marketplace for EpicHero, Jan 17, 2022.
- ❖ Deploy treasury fund and start multichain farming.

Target market and the concept

Target market

- ❖ Anyone who's interested in the Crypto space with long-term investment plans.
- ❖ Anyone who's ready to earn a passive income in BUSD by holding tokens.
- ❖ Anyone who's interested in trading tokens.
- ❖ Anyone who's interested in taking part in staking and farming.
- ❖ Anyone who's interested in taking part with Thoreum multichain venture capital mechanism.
- ❖ Anyone who's interested in taking part with the future plans of the Thoreum V2 token.
- ❖ Anyone who's interested in making financial transactions with any other party using Thoreum V2 or BUSD as the currency.

Core concept

Thoreum V2 reward system

4.5% of each transaction when buying and 9% when selling gets converted to BUSD and is split amongst all holders. Holders will be eligible to receive BUSD when the swap point is reached and rewards are proportional to how many tokens each individual holds.

Sustainable mechanism

The sustainability fee of 4.5% when buying and 9% when selling for treasury/buyback is what allows Thoreum V2 to hold the aforementioned promise. Tokens will be swapped into BNB and will be sent to a wallet when the swap point is reached. This fund will be bridged to multi-chains (such as Avalanche, Polygon, Fantom, etc.) to invest in the most profitable opportunities.

👉Each $Thoreum token represents a share in the profits of Thoreum Multichain Venture Capital.

👉Every time user sell & buy, 6.5% tax is distributed to existing holders in BUSD and 6.5% tax is accumulated in the Thoreum Treasury.

👉 The treasury fund is bridged to different chains and invest in most promising projects (new tokens, NFTs, Metaverses, etc.) and most profitable yield farms.

👉 The profit earned from investing in multi-chain will be used to buyback $Thoreum and distributed in multiple assets to THOREUM holders.

👉 You can just buy the tokens, relax, hold, and reap the earnings from 6.5% tax every transaction & from the buybacks of farming rewards.

✅ **Supply Shock Mechanism:**

3.5% is burnt every buy&sell. This results in a super-fast $Thoreum burn rate.

Because $Thoreum is burned every transaction, its total supply will constantly be deflating against the holder balance, while holder balance is getting the rewards from each transaction and constantly increasing against Thoreum's ever-deflating supply. This built-in mechanism creates a true supply shock to Thoreum token as it becomes ever scarcer with time & volume.

✅ **Ever-increasing utilities:**

Staking: THOREUM is the world first and only token of its kind to offer double staking rewards using our special "Thunder Boost" pools: Place THOREUM in one of the "Thunder Boost" earning pools to earn 3rd-party tokens such as BNB, BUSD, MVC, EpicHero, etc... But the special is here: even now THOREUM is not in your wallet you still continuing to collect your reflection rewards as if the coin is still in your wallet, plus you continue to earn the staking rewards.

NFT Gaming: Thoreum is used in EPICHERO 3D NFT Metaverse - Top 1 Most viewed and Trending Games on Playtoearn.net - as a currency for items, skills, etc.

✅ Driving up scarcity & utilities is the most important task for Thoreum long term's value. With 56,000+ holders and a proven and innovative team behind, expect more utilities/games/metaverse will be rolled out that use Thoreum as part of its mechanism.

❖ The other component of sustainability fee of 1.5% when buying and 3% when the selling will be allocated for marketing, dev, mods, servers, and other costs. This way, Thoreum V2 will have enough funds to promote the coin and spend for future development without selling tokens as the traditional way.

❖ The additional component included under the sustainability section is a **liquidity fee of 1.5% when buying and 3% when selling**, which is a redistribution mechanism that ensures the trading pool always has sufficient liquidity.

❖ Thoreum V2 has a burn strategy that benefits and rewards those who invest long-term. **A 1% fee will be charged when buying and 2% when selling** and sent to the burn wallet. This feature slowly reduces supply making each Thoreum V2 more and more valuable.

Potential to grow with score points

1.	Project efficiency	10/10
2.	Project uniqueness	10/10
3	Information quality	10/10
4	Service quality	10/10
5	System quality	10/10
6	Impact on the community	10/10
7	Impact on the business	10/10
8	Preparing for the future	10/10
Total Points		**10/10**

Contract details

Token contract details for 17th January 2022

Contract name	Thoreum V2 (Thoreum Multi-chain Venture Capital)
Contract address	0xF2A92BC1Cf798fF4De14502a9C6FdA58865e8D5d
Token supply	18,433,655.119423
Token ticker	THOREUM
Decimals	18
Token holders	3,762
Transaction count	5,785
BUSD dividend token	0xe9e7cea3dedca5984780bafc599bd69add087d56
BUSD dividend tracker	0xea5d0d2a24fb0e60d9e915c74c29d6dc0d9a27c0
Marketing wallet	0x8ad9cb111d886dbabbbf232c9a1339b13cb168f8
Treasury wallet	0x312874c97cdd918fa45cd3a3625e012037850ebe
Contract deployer address	0x6A963573b9A7AaEf9CB9EEE747E1b6B2c21b0520
Contract's current owner address	0x6a963573b9a7aaef9cb9eee747e1b6b2c21b0520

Top token holders

Top 10 token holders:
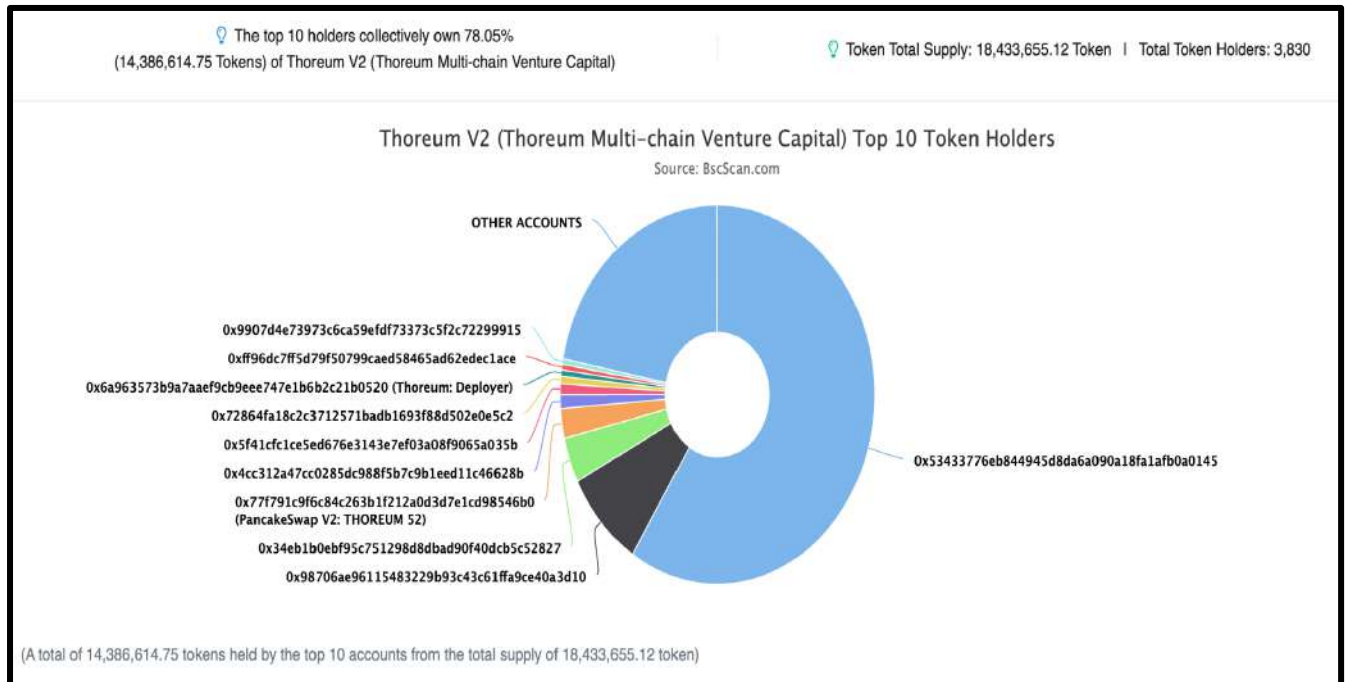
Rank	Address	Quantity (Token)	Percentage
1	📄 0x53433776eb844945d8da6a090a18fa1afb0a0145	10,887,354.669881939409213694	59.0624%
2	📄 0x98706ae96115483229b93c43c61ffa9ce40a3d10	1,545,715	8.3853%
3	0x34eb1b0ebf95c751298d8dbad90f40dcb5c52827	708,134.06	3.8415%
4	📄 PancakeSwap V2: THOREUM 52	468,275.46954191839799906	2.5403%
5	0x4cc312a47cc0285dc988f5b7c9b1eed11c46628b	215,866.09	1.1710%
6	0x5f41cfc1ce5ed676e3143e7ef03a08f9065a035b	170,229.6	0.9235%
7	0x72864fa18c2c3712571badb1693f88d502e0e5c2	118,715	0.6440%
8	Thoreum: Deployer	100,013.857099081303598793	0.5426%
9	0xff96dc7ff5d79f50799caed58465ad62edec1ace	87,657	0.4755%
10	0x9907d4e73973c6ca59efdf73373c5f2c72299915	84,654	0.4592%

Top 100 token holders
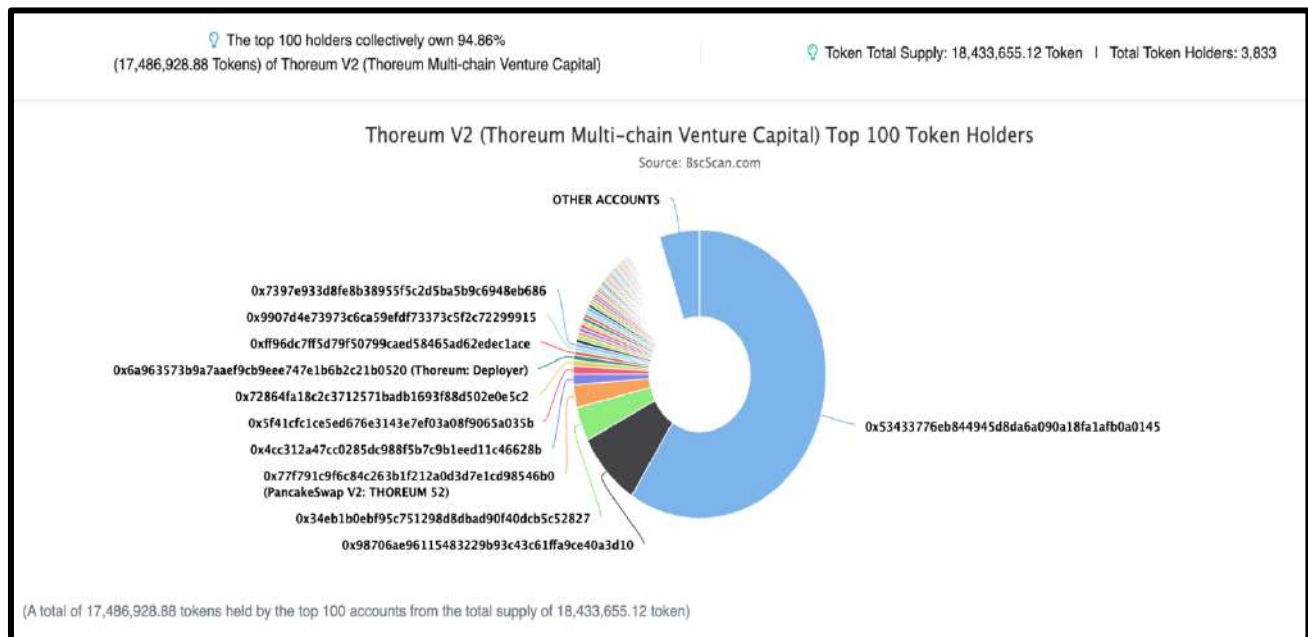
Contract interaction details
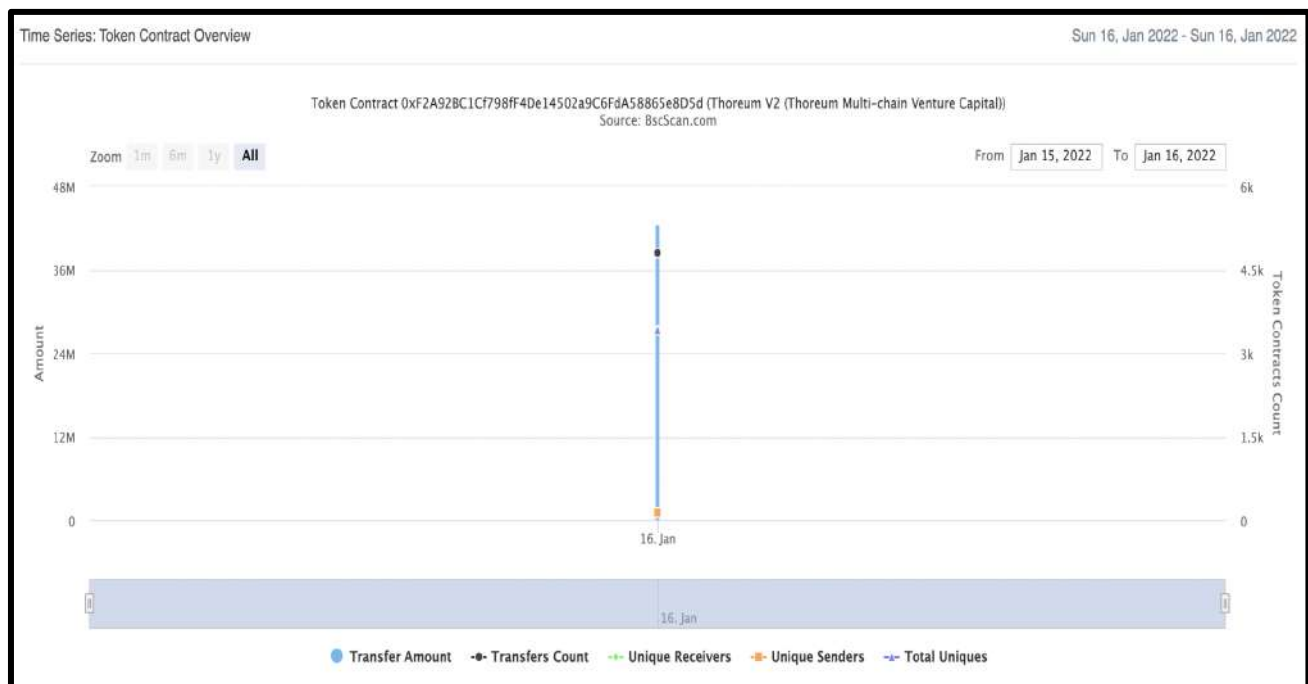
Contract code function details

No	Category	Item	Result
1	Coding conventions	BRC20 Token standards	pass
		compile errors	pass
		Compiler version security	pass
		visibility specifiers	pass
		Gas consumption	pass
		SafeMath features	pass
		Fallback usage	pass
		tx.origin usage	pass
		deprecated items	pass
		Redundant code	pass
		Overriding variables	pass
2	Function call audit	Authorization of function call	pass
		Low level function (call/delegate call) security	pass
		Returned value security	pass
		Selfdestruct function security	pass
3	Business security	Access control of owners	pass
		Business logics	pass
		Business implementations	pass
4	Integer overflow/underflow		pass
5	Reentrancy		pass
6	Exceptional reachable state		pass
7	Transaction ordering dependence		pass
8	Block properties dependence		pass
9	Pseudo random number generator (PRNG)		pass
10	DoS (Denial of Service)		pass
11	Token vesting implementation		pass
12	Fake deposit		pass
13	Event security		pass

Contract description table

Below table represents the summary of the contracts and methods in the token contract. We scanned the whole contract and listed down all the Interfaces, functions and implementations with its visibility and mutability.

Contract	Type	Bases		
L	Function Name	Visibility	Mutability	Modifiers
Context	**Implementation**			
L	_msgSender	Internal 🔒		
L	_msgData	Internal 🔒		
Ownable	**Implementation**	**Context**		
L		Public ❗	🛑	NO❗
L	owner	Public ❗		NO❗
L	renounceOwnership	Public ❗	🛑	onlyOwner
L	transferOwnership	Public ❗	🛑	onlyOwner
L	lock	Public ❗	🛑	onlyOwner
ReentrancyGuard	**Implementation**			
L		Public ❗	🛑	NO❗
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❗
ERC20	**Implementation**	**Context, IERC20**		
L		Public ❗	🛑	NO❗
L	name	Public ❗		NO❗
L	symbol	Public ❗		NO❗
L	decimals	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	decreaseAllowanc e	Public ❗	🛑	NO❗
L	_transfer	Internal 🔒	🛑	
L	_mint	Internal 🔒	🛑	
L	_burn	Internal 🔒	🛑	
L	_approve	Internal 🔒	🛑	
L	_setupDecimals	Internal 🔒	🛑	
L	_beforeTokenTran sfer	Internal 🔒	🛑	
IDEXFactory	**Interface**			
L	createPair	External ❗	🛑	NO❗
IDEXRouter	**Interface**			
L	factory	External ❗		NO❗
L	WETH	External ❗		NO❗
L	addLiquidity	External ❗	🛑	NO❗
L	addLiquidityETH	External ❗	💵	NO❗
L	swapExactTokens ForTokensSupport ingFeeOnTransfer Tokens	External ❗	🛑	NO❗

L	swapExactETHFor TokensSupporting FeeOnTransferTo kens	External ❗	💵	NO❗
L	swapExactTokens ForETHSupporting FeeOnTransferTo kens	External ❗	🛑	NO❗
SafeMath	**Library**			
L	tryAdd	Internal 🔒		
L	trySub	Internal 🔒		
L	tryMul	Internal 🔒		
L	tryDiv	Internal 🔒		
L	tryMod	Internal 🔒		
L	add	Internal 🔒		
L	sub	Internal 🔒		
L	mul	Internal 🔒		
L	div	Internal 🔒		
L	mod	Internal 🔒		
L	sub	Internal 🔒		
L	div	Internal 🔒		
L	mod	Internal 🔒		

SafeMathInt	Library			
L	mul	Internal 🔒		
L	div	Internal 🔒		
L	sub	Internal 🔒		
L	add	Internal 🔒		
L	toUint256Safe	Internal 🔒		
SafeMathUint	Library			
L	toInt256Safe	Internal 🔒		
IterableMapping	Library			
L	get	Public ❗		NO ❗
L	getIndexOfKey	Public ❗		NO ❗
L	getKeyAtIndex	Public ❗		NO ❗
L	size	Public ❗		NO ❗
L	set	Public ❗	🛑	NO ❗
L	remove	Public ❗	🛑	NO ❗
IDividendPayingToken	Interface			
L	dividendOf	External ❗		NO ❗

DividendPayingToken	Implementation	ERC20, IDividendPayingToken		
L	withdrawableDividendOf	External ❗		NO❗
L	withdrawnDividendOf	External ❗		NO❗
L	accumulativeDividendOf	External ❗		NO❗
DividendPayingToken	**Implementation**	**ERC20, IDividendPayingToken**		
L		Public ❗	🛑	ERC20
L		External ❗	💵	NO❗
L	distributeDividends	Public ❗	🛑	NO❗
L	setOnlyCaller	External ❗	🛑	NO❗
L	transferAdmin	External ❗	🛑	NO❗
L	retrieveTokens	External ❗	🛑	NO❗
L	retrieveBNB	External ❗	🛑	NO❗
L	_withdrawDividendOfUser	Internal 🔒	🛑	
L	_claimDividendOfUserToAnyToken	Internal 🔒	🛑	
L	_swapDividendToTokenToAnyToken	Private 🔐	🛑	
L	dividendOf	Public ❗		NO❗
L	withdrawableDividendOf	Public ❗		NO❗
L	withdrawnDividendOf	Public ❗		NO❗

L	accumulativeDividendOf	Public ❗		NO❗
L	_transfer	Internal 🔒	🛑	
L	_mint	Internal 🔒	🛑	
L	_burn	Internal 🔒	🛑	
L	_setBalance	Internal 🔒	🛑	
BusdDividendTracker	**Implementation**	**DividendPayingToken, Ownable**		
L		Public ❗	🛑	DividendPayingToken
L	updateMinimumTokenBalanceForDividends	External ❗	🛑	onlyOwner
L	excludeFromDividends	External ❗	🛑	onlyOwner
L	setBalance	Internal 🔒	🛑	
L	set2Balances	External ❗	🛑	onlyOwner
L	claimDividend	Public ❗	🛑	onlyOwner
L	compoundDividend	Public ❗	🛑	onlyOwner
L	_transfer	Internal 🔒		
L	getNumberOfTokenHolders	External ❗		NO❗
ThoreumV2	**Implementation**	**ERC20, Ownable, ReentrancyGuard**		

L		Public !	⬤	ERC20
L		External !	💵	NO!
L	prepareForPartner	External !	⬤	onlyOwner
L	setMaxBuyTransaction	External !	⬤	onlyOwner
L	setMaxSellTransaction	External !	⬤	onlyOwner
L	setBusdDividendToken	External !	⬤	onlyOwner
L	setMarketingWallet	External !	⬤	onlyOwner
L	setTreasuryWallet	External !	⬤	onlyOwner
L	setToBurnTokenFee	External !	⬤	onlyOwner
L	setMaxWalletToken	External !	⬤	onlyOwner
L	setSwapTokensAtAmount	External !	⬤	onlyOwner
L	setSellTransactionMultiplier	External !	⬤	onlyOwner
L	setIsNotMigrating	External !	⬤	onlyOwner
L	setIsLiquidityInBnb	External !	⬤	onlyOwner
L	setBusdDividendTracker	External !	⬤	onlyOwner
L	setBusdDividendRewardFee	External !	⬤	onlyOwner
L	setMarketingFee	External !	⬤	onlyOwner
L	setTreasuryFee	External !	⬤	onlyOwner

L	setLiquidityFee	External ❗	🛑	onlyOwner
L	setFeesOnNormal Transfers	External ❗	🛑	onlyOwner
L	setAutomatedMark etMakerPair	Public ❗	🛑	onlyOwner
L	_setAutomatedMa rketMakerPair	Private 🔒	🛑	onlyOwner
L	setMinimumBalan ceForDividends	External ❗	🛑	onlyOwner
L	excludeFromDivid end	Public ❗	🛑	onlyOwner
L	claim	External ❗	🛑	nonReentr ant
L	compound	External ❗	🛑	nonReentr ant
L	excludeFromFees	Public ❗	🛑	onlyOwner
L	retrieveTokens	External ❗	🛑	onlyOwner
L	retrieveBNB	External ❗	🛑	onlyOwner
L	_updateTotalFee	Internal 🔒	🛑	
L	_transfer	Internal 🔒	🛑	
L	_swapAndLiquify	Private 🔒	🛑	
L	_addLiquidity	Private 🔒	🛑	
L	_swapTokensForB NB	Private 🔒	🛑	
L	_addLiquidityBusd	Private 🔒	🛑	
L	_swapTokensForB usd	Private 🔒	🛑	

L	_swapTokensForDividendToken	Private 🔒	🛑	
L	_swapAndSendBusdDividends	Private 🔒	🛑	
L	_transferToWallet	Private 🔒	🛑	

Legend

Symbol	Meaning
🛑	**Function can modify state**
💵	**Function is payable**

Inheritance Hierarchy
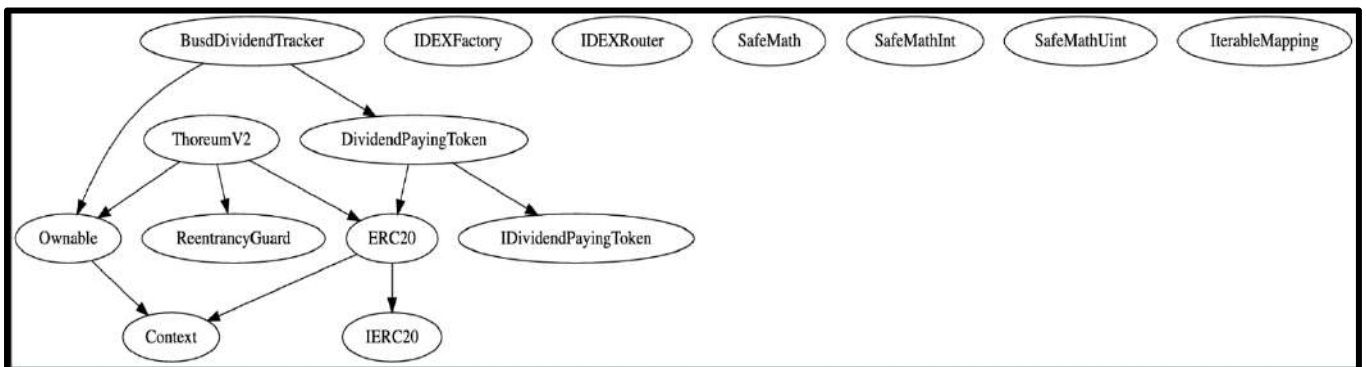
Security issue checking status

❖ **High severity issues**

No high severity issues found.

❖ **Medium severity issues**

No medium severity issues found.

❖ **Low severity issues**

No low severity issues found.

❖ **Informational**

- The owner can change max buy, sell transactions, and wallet limit without minimum value condition.

```
ftrace | funcSig
function setMaxBuyTransaction(uint256 _maxTxn↑) external onlyOwner {
    maxBuyTransactionAmount = _maxTxn↑;
}


ftrace | funcSig
function setMaxSellTransaction(uint256 _maxTxn↑) external onlyOwner {
    maxSellTransactionAmount = _maxTxn↑;
}
```

```
ftrace | funcSig
function setMaxWalletToken(uint256 _maxToken↑) external onlyOwner {
    maxWalletToken = _maxToken↑;
}
```

- The owner can change the sell fee multiplier without any max limit.

```
ftrace | funcSig
function setSellTransactionMultiplier(uint256 _multiplier↑)
    external
    onlyOwner
{
    sellFeeIncreaseFactor = _multiplier↑;
}
```

Owner privileges

❖ The owner can change max sell and buy transactions limit.

```
ftrace | funcSig
function setMaxBuyTransaction(uint256 _maxTxn↑) external onlyOwner {
    maxBuyTransactionAmount = _maxTxn↑;
}

ftrace | funcSig
function setMaxSellTransaction(uint256 _maxTxn↑) external onlyOwner {
    maxSellTransactionAmount = _maxTxn↑;
}
```

❖ The owner can change reward token address.

```
ftrace | funcSig
function setBusdDividendToken(address _newContract↑) external onlyOwner {
    busdDividendToken = _newContract↑;
    IERC20(busdDividendToken).approve(address(dexRouter), 2**256 - 1);
}
```

❖ The owner can change marketing and treasury wallet address.

```
ftrace | funcSig
function setMarketingWallet(address _newWallet↑) external onlyOwner {
    excludeFromFees(_newWallet↑, true);
    marketingWallet = _newWallet↑;
}

ftrace | funcSig
function setTreasuryWallet(address _newWallet↑) external onlyOwner {
    excludeFromFees(_newWallet↑, true);
    treasuryWallet = _newWallet↑;
}
```

❖ The owner can change burn fee maximum up to 10%.

```
ftrace | funcSig
function setToBurnTokenFee(uint256 newFee↑) external onlyOwner {
    require(newFee↑ <= MAX_FEE_RATE, "wrong");
    toBurnTokenFee = newFee↑;
    _updateTotalFee();
}
```

❖ The owner can change max wallet token amount.

```
ftrace | funcSig
function setMaxWalletToken(uint256 _maxToken↑) external onlyOwner {
    maxWalletToken = _maxToken↑;
}
```

❖ The owner can change token swap point.

```
ftrace | funcSig
function setSwapTokensAtAmount(uint256 _swapAmount↑) external onlyOwner {
    swapTokensAtAmount = _swapAmount↑;
}

ftrace | funcSig
```

❖ The owner can change sell fee multiplier.

```
ftrace | funcSig
function setSellTransactionMultiplier(uint256 _multiplier↑)
    external
    onlyOwner
{
    sellFeeIncreaseFactor = _multiplier↑;
}
```

❖ The owner can change migrating status (if isNotMigrating values false then users cannot interact with the contract)

```
ftrace | funcSig
function setIsNotMigrating(bool _value↑) external onlyOwner {
    require(isNotMigrating != _value↑, "Not changed");
    isNotMigrating = _value↑;
}
```

❖ The owner can change the dividend tracker.

```
ftrace | funcSig
function setBusdDividendTracker(address newAddress↑) external onlyOwner {
    BusdDividendTracker newBusdDividendTracker = BusdDividendTracker(
        payable(newAddress↑)
    );

    require(
        newBusdDividendTracker.owner() == address(this),
        "must be owned by Thoreum"
    );

    newBusdDividendTracker.excludeFromDividends(
        address(newBusdDividendTracker),
        true
    );
    newBusdDividendTracker.excludeFromDividends(address(this), true);
    newBusdDividendTracker.excludeFromDividends(address(dexRouter), true);
    newBusdDividendTracker.excludeFromDividends(address(dexPair), true);
    newBusdDividendTracker.excludeFromDividends(address(dexPairBusd), true);
    newBusdDividendTracker.excludeFromDividends(address(deadAddress), true);

    busdDividendTracker = newBusdDividendTracker;
}
```

❖ The owner can enable/disable fees on normal transactions.

```
ftrace | funcSig
function setFeesOnNormalTransfers(bool _enabled↑) external onlyOwner {
    feesOnNormalTransfers = _enabled↑;
}
```

❖ The owner can change the minimum token balance to have dividends.

```
ftrace | funcSig
function setMinimumBalanceForDividends(uint256 newMinimumBalance↑)
    external
    onlyOwner
{
    busdDividendTracker.updateMinimumTokenBalanceForDividends(
        newMinimumBalance↑
    );
}
```

❖ The owner can exclude wallets from dividends.

```
ftrace | funcSig
function excludeFromDividend(address account↑, bool status↑)
    public
    onlyOwner
{
    busdDividendTracker.excludeFromDividends(address(account↑), status↑);
}
```

❖ The owner can exclude wallets from fees.

```
ftrace | funcSig
function excludeFromFees(address account↑, bool excluded↑) public onlyOwner {
    require(isExcludedFromFees[account↑] != excluded↑, "Already excluded");
    isExcludedFromFees[account↑] = excluded↑;

    emit ExcludeFromFees(account↑, excluded↑);
}
```

❖ The owner can get all tokens and BNB inside the contract to owner's wallet.

```
ftrace | funcSig
function retrieveTokens(address token↑) external onlyOwner {
    uint256 amount = IERC20(token↑).balanceOf(address(this));

    require(IERC20(token↑).transfer(msg.sender, amount), "Transfer failed");
}

ftrace | funcSig
function retrieveBNB() external onlyOwner {
    uint256 amount = address(this).balance;

    (bool success, ) = payable(msg.sender).call{value: amount}("");
    require(success, "Failed to retrieve BNB");
}
```

❖ The owner can change all fees (one fee maximum up to 10%)

```
ftrace | funcSig
function setBusdDividendRewardFee(uint256 newFee) external onlyOwner {
    require(newFee <= MAX_FEE_RATE, "wrong");
    busdDividendRewardsFee = newFee;
    _updateTotalFee();
}


ftrace | funcSig
function setMarketingFee(uint256 newFee) external onlyOwner {
    require(newFee <= MAX_FEE_RATE, "wrong");
    marketingFee = newFee;
    _updateTotalFee();
}


ftrace | funcSig
function setTreasuryFee(uint256 newFee) external onlyOwner {
    require(newFee <= MAX_FEE_RATE, "wrong");
    treasuryFee = newFee;
    _updateTotalFee();
}


ftrace | funcSig
function setLiquidityFee(uint256 newFee) external onlyOwner {
    require(newFee <= MAX_FEE_RATE, "wrong");
    liquidityFee = newFee;
    _updateTotalFee();
}
```

Audit conclusion

While conducting the audit of the Thoreum V2 smart contract, it was observed that there is nothing alarming with the code.