RugFreeCoins Audit

Wings Protocol

Smart Contract Security Audit

February 09, 2022

Contents

Audit details

Audited project

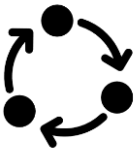
Wings Protocol Token

Contract Address

0xc0fe33B654d13AF5a72C47Dc5a370674ba85b3E6

Client contact

Wings Protocol Team



Blockchain

Binance smart chain

Project website

https://www.wingsprotocol.com/

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 Wings Protocol Token to perform an audit of the smart contract.

https://bscscan.com/token/0xc0fe33B654d13AF5a72C47Dc5a370674ba85b3E6

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

Wings Protocol 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 even greater Community, Multi Chain NFT marketplace, staking platform, launchpad and p2e games. Each transaction, purchase incurs 8% fee, and sale incurs a 12% fee.

Tokenomics

8% fee when buying

- 3% of trade goes to project funds wallet in BNB
- 2% of trade goes to the liquidity pool.
- 1% of trade goes to the manual buyback wallet in BNB
- 2% of trade goes to the staking wallet in tokens.

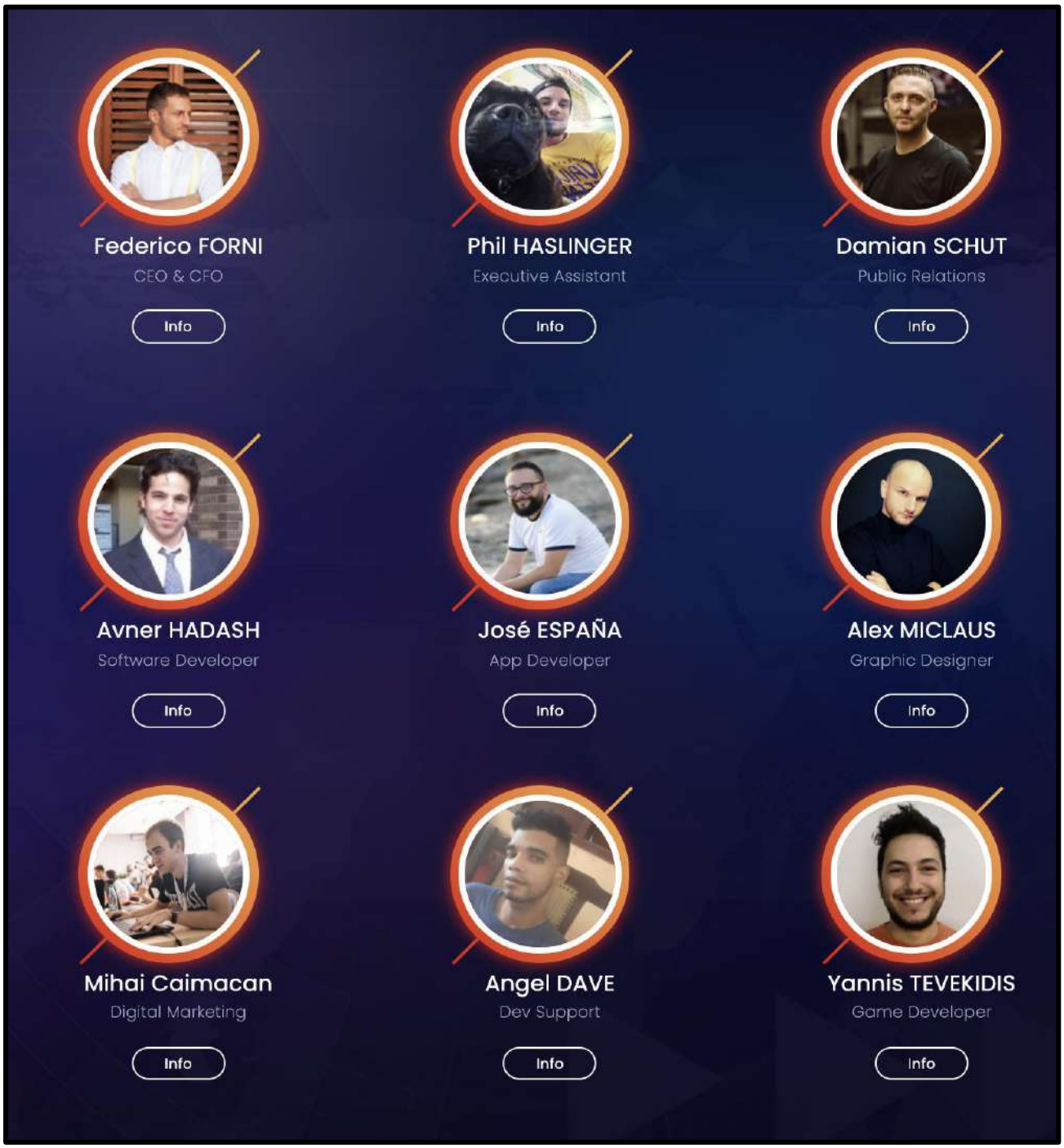
12% fee when buying

- 6% of trade goes to project funds wallet in BNB
- 2% of trade goes to the liquidity pool.
- 2% of trade goes to the manual buyback wallet in BNB.
- 2% of trade goes to the staking wallet in tokens.

Roadmap


PHASE 1
2021
Structuring
WhitePaper
Internal Audit
Staking Development
MarketPlace Development
PHASE 2
2022
Private Sale
Migration Launch
LaunchPad Product
External Audit
NFTs Drop
iOS Application Launch
Game Development
PHASE 3
2022
LCEX Listing
Game Launch

Team


Federico FORNI
CEO & CFO
Info
Phil HASLINGER
Executive Assistant
Info
Damian SCHUT
Public Relations
Info
Avner HADASH
Software Developer
Info
José ESPAÑA
App Developer
Info
Alex MICLAUS
Graphic Designer
Info
Mihai Caimacan
Digital Marketing
Info
Angel DAVE
Dev Support
Info
Yannis TEVEKIDIS
Game Developer
Info

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 by holding tokens.
- Anyone who's interested in trading tokens.
- Anyone who's interested in collecting NFTs or trading NFTs.
- Anyone who's interested in taking part with staking.
- Anyone who's interested in taking part with launchpad.
- Anyone who's interested in taking part with the future plans of the Wings Protocol token.
- Anyone who's interested in making financial transactions with any other party using Wings Protocol as the currency.

Core concept

Sustainable mechanism

The **sustainability fee of 3% when buying and 6% when selling for project funds** is what allows Wings Protocol Token to promote the token and use funds to further the development of the platform. Tokens will be swapped into BNB and will be sent to a project fund wallet. This way, Wings Protocol Token will have access to the funds without selling tokens as the traditional way, which will enable them to consume funds without hurting the project.

The fee of **1% when buying and 2% when selling for manual buyback** is what allows Wings Protocol Token to buyback tokens manually when the market is at a dip.

The liquidity fee of 2% when buying and selling, which is a redistribution mechanism that ensures the trading pool always has sufficient liquidity.

The fee of 2% for staking when buying and selling is what keep pouring tokens to the staking pool for staking rewards.

Staking

There is a way to reap the rewards of mining without investing in expensive hardware or worrying about maintenance. It is cryptocurrency staking, and it is a convenient way to generate passive income.

Staking offers an eco-friendly, energy-efficient alternative to mining. A PoS blockchain requires volunteers to commit their crypto assets for the validation process.

If you have xWIP sitting idle, staking may be an option to earn some passive income. It is the same as earning interest on a fixed deposit but with the potential for higher interest/rewards. Our attractive APYs can vary according to the amount of stakers. Our tokenomics reserve a portion of the transaction taxes to sustain the staking rewards alongside with a dedicated staking supply.

LaunchPad

This is a central feature in the WINGS environment. Quality projects will be deployed through Wings Protocol accordingly, and the token xWIP will give their holders guaranteed pre allocations (with parameters accurately set, e.g. amount of holding, hard cap for each private sale, etc.) Therefore, by simply holding an amount of xWIP tokens, members can become early investors in promising projects that will launch through the platform.

Web 3.0 is the present and the future.

Web 3.0 is slated to be the new paradigm in web interaction and will mark a fundamental change in how people connect with websites and crypto.

Marketplace

Another active feature of the ecosystem will be the Marketplace, where artists can be met and everyone could create and sell non-fungible tokens (NFTs) and other crypto related products.

The 1st NFT collection to be launched through the platform will be “Digital Rebels”, revisiting Rebels from all around beliefs, anime, cinema and history.
“Digital Rebels” are divided into tiers, of which the 1st will have the most valuable in matter of price and benefits for the IDOs.

Gaming

A branch of the platform will be focused on mini games and play-to-earn (p2e) games. GameFi is one of the hottest new trends within the crypto currencies cosmo, combining decentralized finance (DeFi) and NFTs with blockchain-based games.

The ‘Play-to-earn’ concept involves players giving them financial incentives to play and progress through the games.

As recent history teaches, it’s known that this has allowed gamers to earn a full-time income just by playing.

News

To stay one step ahead of the competition it requires a high level of information.

Information and knowledge mean power to make the smartest decision in terms of assets management.

To help our members in making clever moves on the market the Wings Protocol will have partnerships with the most prestigious news channels in order to keep every user up-to-date with the latest trends within the crypto market.

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**

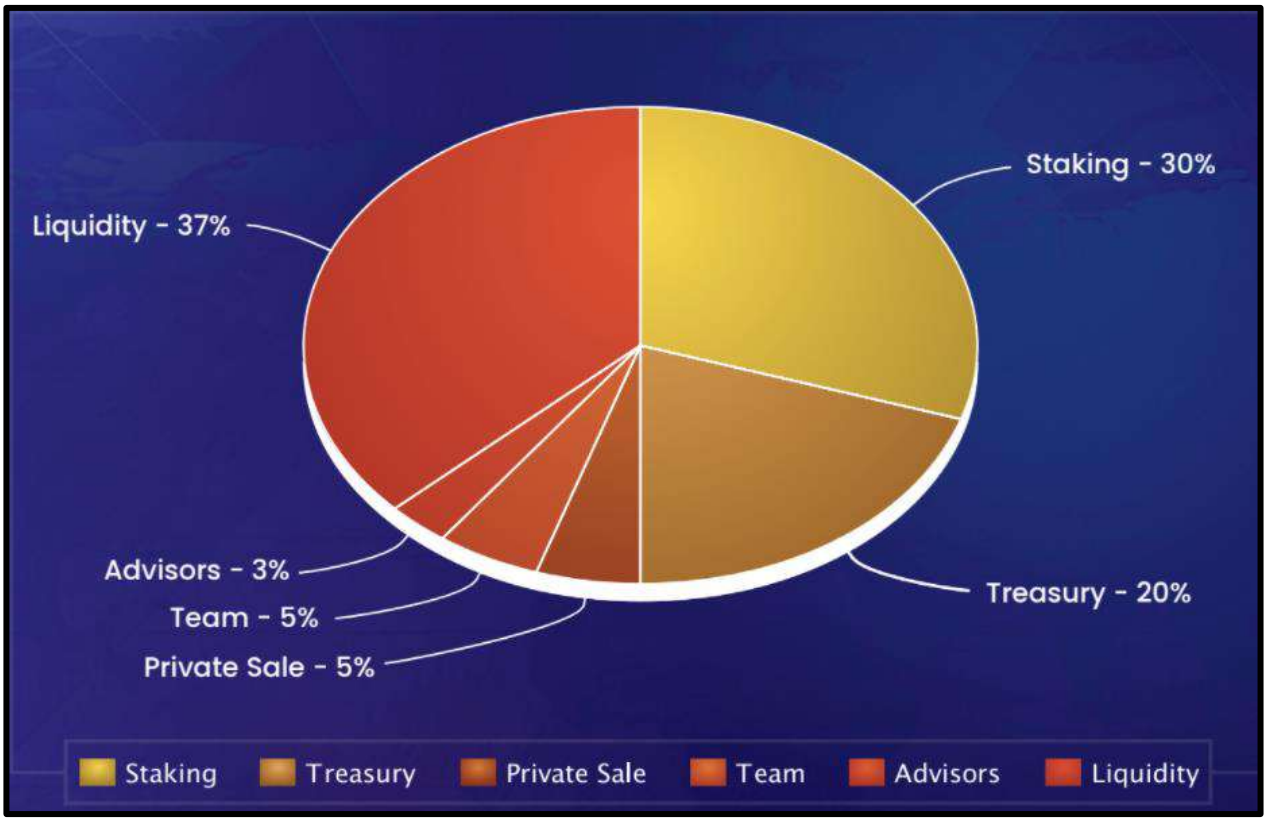
Contract details

Token contract details for 09th February 2022

Contract name	WingsProtocol
Contract address	0xc0fe33B654d13AF5a72C47Dc5a370674ba85b3E6
Token supply	100,000,000,000
Token ticker	XWIP
Decimals	18
Token holders	1
Transaction count	2
Buyback wallet	0x33a2318f9da904bffc3375c3a264e055e40f30ee
Project fund receiver wallet	0xfde9b2aa3810007e6d6ca4648976d3d2e7262c7f
Staking fee receiver wallet	0x6378633a8adedb70593369a5e7ab37662bd2a76f
Contract deployer address	0x97361F6979756A647458DC3EAAB802Db416997a3
Contract’s current owner address	0x7f37f6b06f98f2bdd808cd58a1cd08c33a7dea1f

Token distribution

Tokens are distributed as follows



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
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❗
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❗

L	swapExactTokensForTokensSupportingFeeOnTransferTokens	External ❗	🛑	NO❗
L	swapExactETHForTokensSupportingFeeOnTransferTokens	External ❗	🗜	NO❗
L	swapExactTokensForETHSupportingFeeOnTransferTokens	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 🔒		

IERC20	Interface			
└	totalSupply	External ❗		NO❗
└	balanceOf	External ❗		NO❗
└	transfer	External ❗	🛑	NO❗
└	allowance	External ❗		NO❗
└	approve	External ❗	🛑	NO❗
└	transferFrom	External ❗	🛑	NO❗
Context	**Implementation**			
└	_msgSender	Internal 🔒		
└	_msgData	Internal 🔒		
Ownable	**Implementation**	**Context**		
└		Public ❗	🛑	NO❗
└	owner	Public ❗		NO❗
└	renounceOwnership	Public ❗	🛑	onlyOwner
└	transferOwnership	Public ❗	🛑	onlyOwner
└	_transferOwnership	Internal 🔒	🛑	
WingsProtocol	**Implementation**	**IERC20, Ownable**		
└		Public ❗	🛑	NO❗
└		External ❗	💵	NO❗

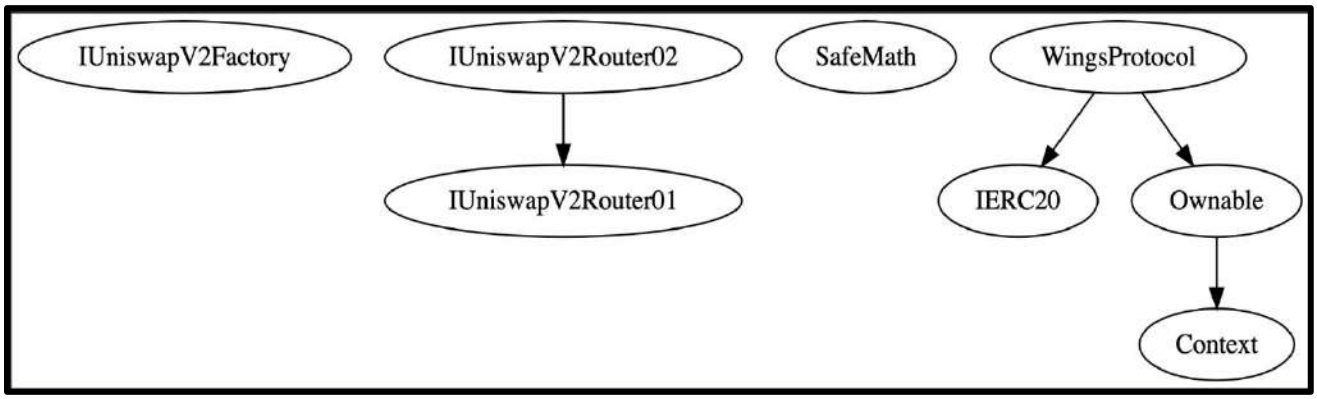
└	totalSupply	External ❗		NO❗
└	name	Public ❗		NO❗
└	symbol	Public ❗		NO❗
└	decimals	Public ❗		NO❗
└	balanceOf	Public ❗		NO❗
└	allowance	External ❗		NO❗
└	approve	Public ❗	🛑	NO❗
└	_approve	Internal 🔒	🛑	
└	approveMax	External ❗	🛑	NO❗
└	transfer	External ❗	🛑	NO❗
└	transferFrom	External ❗	🛑	NO❗
└	_transferFrom	Internal 🔒	🛑	
└	_basicTransfer	Internal 🔒	🛑	
└	shouldTakeFee	Internal 🔒		
└	takeFee	Internal 🔒	🛑	
└	shouldSwapBack	Internal 🔒		
└	clearStuckBalance	External ❗	🛑	onlyOwner
└	updateUniswapRouter	Public ❗	🛑	onlyOwner
└	updateBuyFees	Public ❗	🛑	onlyOwner
└	updateSellFees	Public ❗	🛑	onlyOwner

L	changeSwapAmount	Public ❗	🛑	onlyOwner
L	tradingStatus	Public ❗	🛑	onlyOwner
L	setGetFeesOnNormalTx	Public ❗	🛑	onlyOwner
L	setIsAuthorize	Public ❗	🛑	onlyOwner
L	blackListWallets	Public ❗	🛑	onlyOwner
L	exemptTimeLock	Public ❗	🛑	onlyOwner
L	exemptMaxWallet	Public ❗	🛑	onlyOwner
L	changeMaxWalletToken	Public ❗	🛑	onlyOwner
L	enableSellCollDown	Public ❗	🛑	onlyOwner
L	changeSellCoolDownTime	Public ❗	🛑	onlyOwner
L	swapBackInBnb	Internal 🔒	🛑	swapping
L	swapAndLiquify	Private 🔐	🛑	
L	swapTokensForEth	Private 🔐	🛑	
L	addLiquidity	Private 🔐	🛑	
L	setIsFeeExempt	External ❗	🛑	onlyOwner
L	setFeeReceivers	External ❗	🛑	onlyOwner
L	setSwapBackSettings	External ❗	🛑	onlyOwner

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**

  - Owner can enable/disable trading anytime.

```
ftrace | funcSig
function tradingStatus(bool _status) public onlyOwner {
    tradingOpen = _status;
}
```

Owner privileges

- The owner can update router address.

```
function updateUniswapRouter(address newAddress) public onlyOwner {
    require(
        newAddress != address(router),
        "The router already has that address"
    );

    router = IUniswapV2Router02(newAddress);
    pair = IUniswapV2Factory(router.factory()).createPair(
        WBNB,
        address(this)
    );
}
```

- The owner can change token swap percentages.

```
function changeSwapAmount(
    uint256 _buyBackSwap,
    uint256 _liquiditySwap,
    uint256 _projectSwap
) public onlyOwner {
    buyBackSwap = _buyBackSwap;
    liquiditySwap = _liquiditySwap;
    projectSwap = _projectSwap;
    uint256 totalSwap = _buyBackSwap.add(_liquiditySwap).add(_projectSwap);

    require(totalSwap == 100, "Total swap percentage should equal to 100%");
}
```

- The owner can get contract BNB balance to the owner wallet.

```
ftrace | funcSig
function clearStuckBalance(uint256 amountPercentage) external onlyOwner {
    uint256 amountBNB = address(this).balance;
    payable(msg.sender).transfer((amountBNB * amountPercentage) / 100);
}
```

❖ The owner can change all buy and sell fees.

```
ftrace | funcSig
function updateBuyFees(
    uint256 buyBack⬆,
    uint256 projectFee⬆,
    uint256 liquidity⬆,
    uint256 staking⬆
) public onlyOwner {
    buyBuyBackFee = buyBack⬆;
    buyProjectFundFee = projectFee⬆;
    buyLiquidityFee = liquidity⬆;
    buyStakingPoolFee = staking⬆;
    buyTotalFees = buyBack⬆.add(projectFee⬆).add(liquidity⬆).add(staking⬆);
}

ftrace | funcSig
function updateSellFees(
    uint256 buyBack⬆,
    uint256 projectFee⬆,
    uint256 liquidity⬆,
    uint256 staking⬆
) public onlyOwner {
    sellBuyBackFee = buyBack⬆;
    sellProjectFundFee = projectFee⬆;
    sellLiquidityFee = liquidity⬆;
    sellStakingPoolFee = staking⬆;
    sellTotalFees = buyBack⬆.add(projectFee⬆).add(liquidity⬆).add(staking⬆);
}
```

❖ The owner can enable/disable trading.

```
// switch Trading
ftrace | funcSig
function tradingStatus(bool _status⬆) public onlyOwner {
    tradingOpen = _status⬆;
}
```

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

```
ftrace | funcSig
function setGetFeesOnNormalTx(bool _status⬆) public onlyOwner {
    isGetNormalFees = _status⬆;
}
```

❖ The owner can add/remove wallets from authorizations.

```
ftrace | funcSig
function setIsAuthorize(address wallet⬆, bool _status⬆) public onlyOwner {
    isAuthorized[wallet⬆] = _status⬆;
}
```

❖ The owner can add/remove wallets from blacklist.

```
ftrace | funcSig
function blackListWallets(address wallet⬆, bool _status⬆) public onlyOwner {
    isBlacklist[wallet⬆] = _status⬆;
}
```

❖ The owner can exclude wallets from timelock and max wallet tokens.

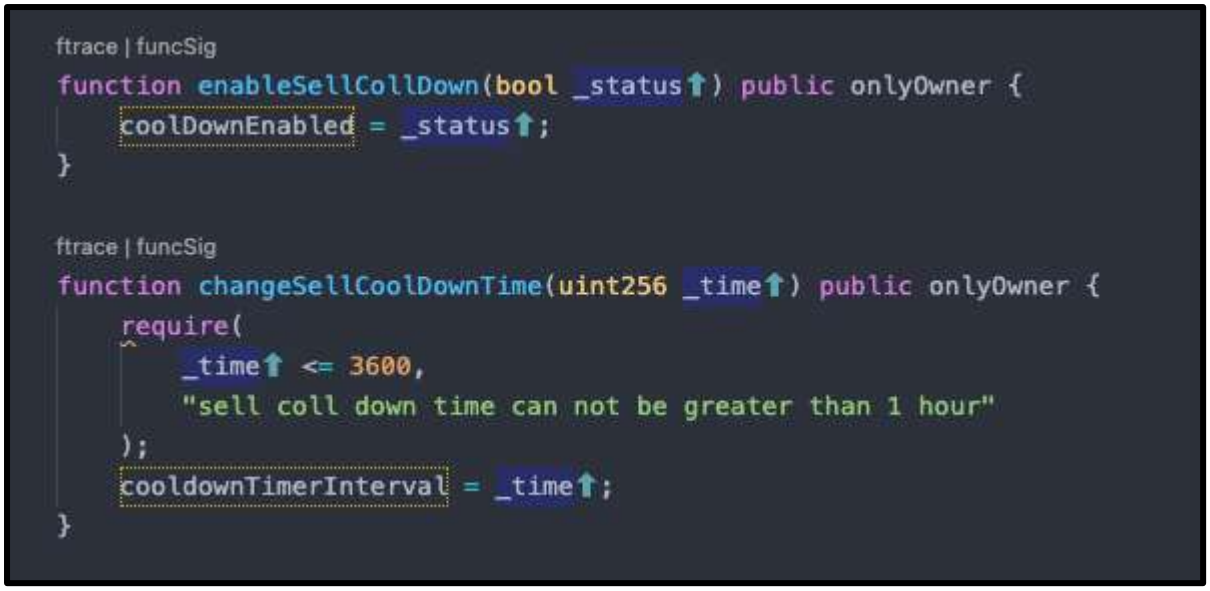
```
ftrace | funcSig
function exemptTimeLock(address wallet⬆, bool _status⬆) public onlyOwner {
    isTimelockExempt[wallet⬆] = _status⬆;
}

ftrace | funcSig
function exemptMaxWallet(address wallet⬆, bool _status⬆) public onlyOwner {
    isMaxWalletExempt[wallet⬆] = _status⬆;
}
```

❖ The owner can change max wallet token minimum up to 1000000000.

```
ftrace | funcSig
function changeMaxWalletToken(uint256 _amount⬆) public onlyOwner {
    require(
        _amount⬆ >= 1000000000,
        "Max wallet amount can not be less than 1%"
    );
    _maxWalletToken = _amount⬆ * 10**_decimals;
}
```

- The owner can enable/disable sell cooldown and can change cool down time maximum upto 1 hour.



```
ftrace | funcSig
function enableSellCollDown(bool _status⬆) public onlyOwner {
    coolDownEnabled = _status⬆;
}

ftrace | funcSig
function changeSellCoolDownTime(uint256 _time⬆) public onlyOwner {
    require(
        _time⬆ <= 3600,
        "sell coll down time can not be greater than 1 hour"
    );
    cooldownTimerInterval = _time⬆;
}
```

- The owner can exclude wallets from fees.

```
ftrace | funcSig
function setIsFeeExempt(address holder⬆, bool exempt⬆) external onlyOwner {
    isFeeExempt[holder⬆] = exempt⬆;
}
```

- The owner can change fee receivers.

```
ftrace | funcSig
function setFeeReceivers(
    address _buyBackWallet⬆,
    address _stakingWallet⬆,
    address _projectFundWallet⬆
) external onlyOwner {
    buyBackFeeReceiver = _buyBackWallet⬆;
    stakingPoolFeeReceiver = _stakingWallet⬆;
    projectFundFeeReceiver = _projectFundWallet⬆;
}
```

❖ The owner can change swap back settings.

Audit conclusion

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