expelee

Building the Futuristic Blockchain Ecosystem

SECURITY AUDIT REPORT

NINJA

TOKEN OVERVIEW

Risk Findings

Severity	Found
🔴 High	2
🟠 Medium	1
🟡 Low	0
🔵 Informational	0

Centralization Risks

Owner Privileges	Description
🟢 Can Owner Set Taxes >25% ?	Not Detected
🟢 Owner needs to enable trading ?	Not Detected
🟢 Can Owner Disable Trades ?	Not Detected
🟢 Can Owner Mint ?	Not Detected
🟢 Can Owner Blacklist ?	Not Detected
🟢 Can Owner set Max Wallet amount ?	Not Detected
🟢 Can Owner Set Max TX amount ?	Not Detected

TABLE OF CONTENTS

OVERVIEW

The Expelee team has performed a line-by-line manual analysis and automated review of the smart contract. The smart contract was analysed mainly for common smart contract vulnerabilities, exploits, and manipulation hacks. According to the smart contract audit:

Audit Result	**Passed With High Risk**
KYC Verification	-
Audit Date	**22 July 2023**

CONTRACT DETAILS

Token Name: Ninja

Symbol: NINJA

Network: Binance Smart Chain

Language: Solidity

Contract Address:
0xf8173DfE1998265016eED56EE9B5d8E988b57ca6

Total Supply: 100,000,000,000

Owner's Wallet:
0x9e84d30c449889500710F436eAeBC8F732ac459d

Deployer's Wallet:
0x9e84d30c449889500710F436eAeBC8F732ac459d

Testnet.
https://testnet.bscscan.com/address/0xD75B85108ec8741fCDA5d9C951bC3703241C0322

AUDIT METHODOLOGY

Audit Details

Our comprehensive audit report provides a full overview of the audited system's architecture, smart contract codebase, and details on any vulnerabilities found within the system.

Audit Goals

The audit goal is to ensure that the project is built to protect investors and users, preventing potentially catastrophic vulnerabilities after launch , that lead to scams and rugpulls.

Code Quality

Our analysis includes both automatic tests and manual code analysis for the following aspects:

- Exploits
- Back-doors
- Vulnerability
- Accuracy
- Readability

Tools

- DE
- Open Zeppelin
- Code Analyzer
- Solidity Code
- Compiler
- Hardhat

VULNERABILITY CHECKS

Design Logic	Passed
Compiler warnings	Passed
Private user data leaks	Passed
Timestamps dependence	Passed
Integer overflow and underflow	Passed
Race conditions & reentrancy. Cross-function race conditions	Passed
Possible delays in data delivery	Passed
Oracle calls	Passed
Front Running	Passed
DoS with Revert	Passed
DoS with block gas limit	Passed
Methods execution permissions	Passed
Economy model	Passed
Impact of the exchange rate on the logic	Passed
Malicious event log	Passed
Scoping and declarations	Passed
Uninitialized storage pointers	Passed
Arithmetic accuracy	Passed
Cross-function race conditions	Passed
Safe Zepplin module	Passed

RISK CLASSIFICATION

When performing smart contract audits, our specialists look for known vulnerabilities as well as logical and acces control issues within the code. The exploitation of these issues by malicious actors may cause serious financial damage to projects that failed to get an audit in time. We categorize these vulnerabilities by the following levels:

High Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

Medium Risk

Issues on this level are critical to the smart contract's performance/functionality and should be fixed before moving to a live environment.

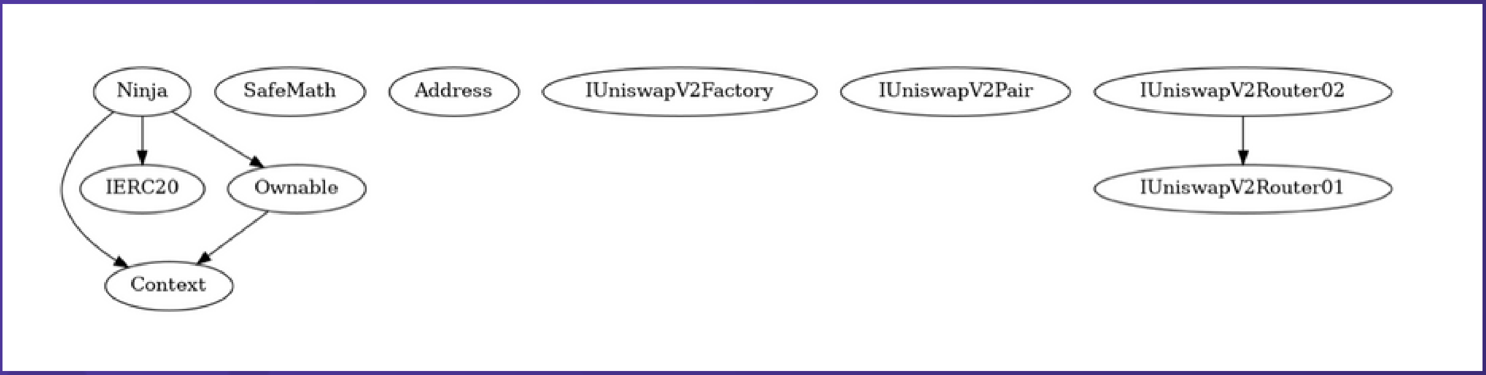
Low Risk

Issues on this level are minor details and warning that can remain unfixed.

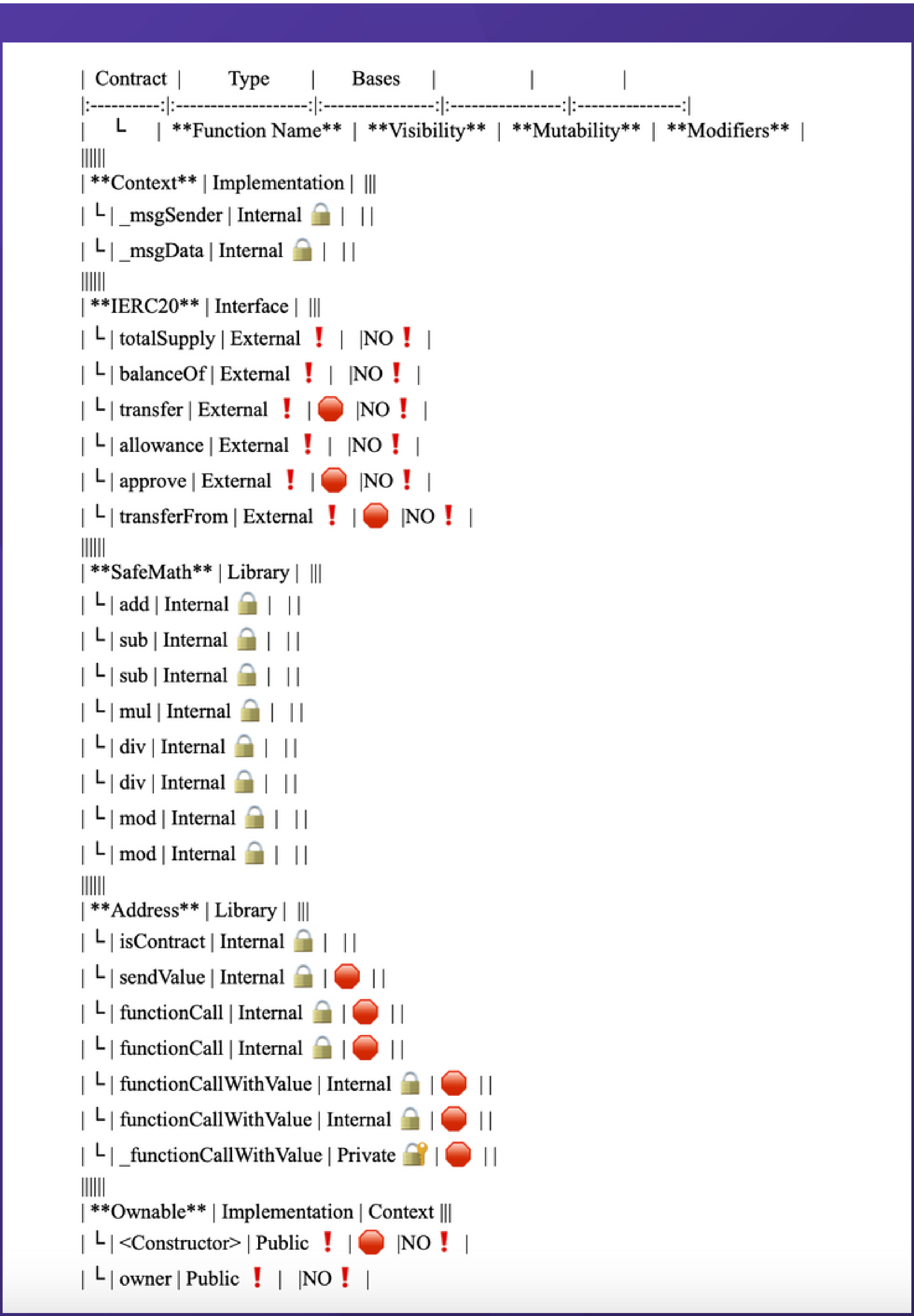
Informational

Issues on this level are minor details and warning that can remain unfixed.

INHERITANCE TREES

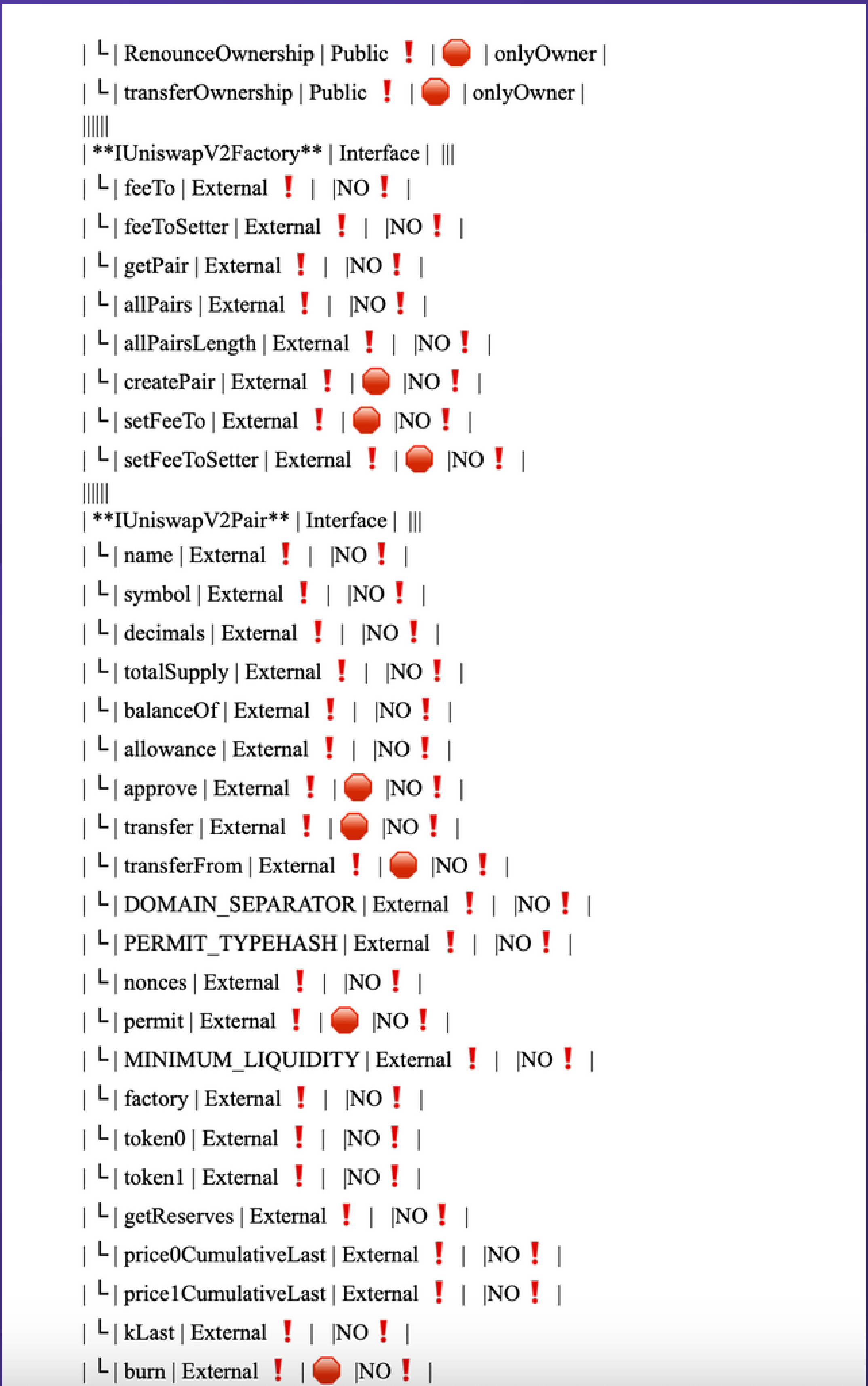


FUNCTION DETAILS



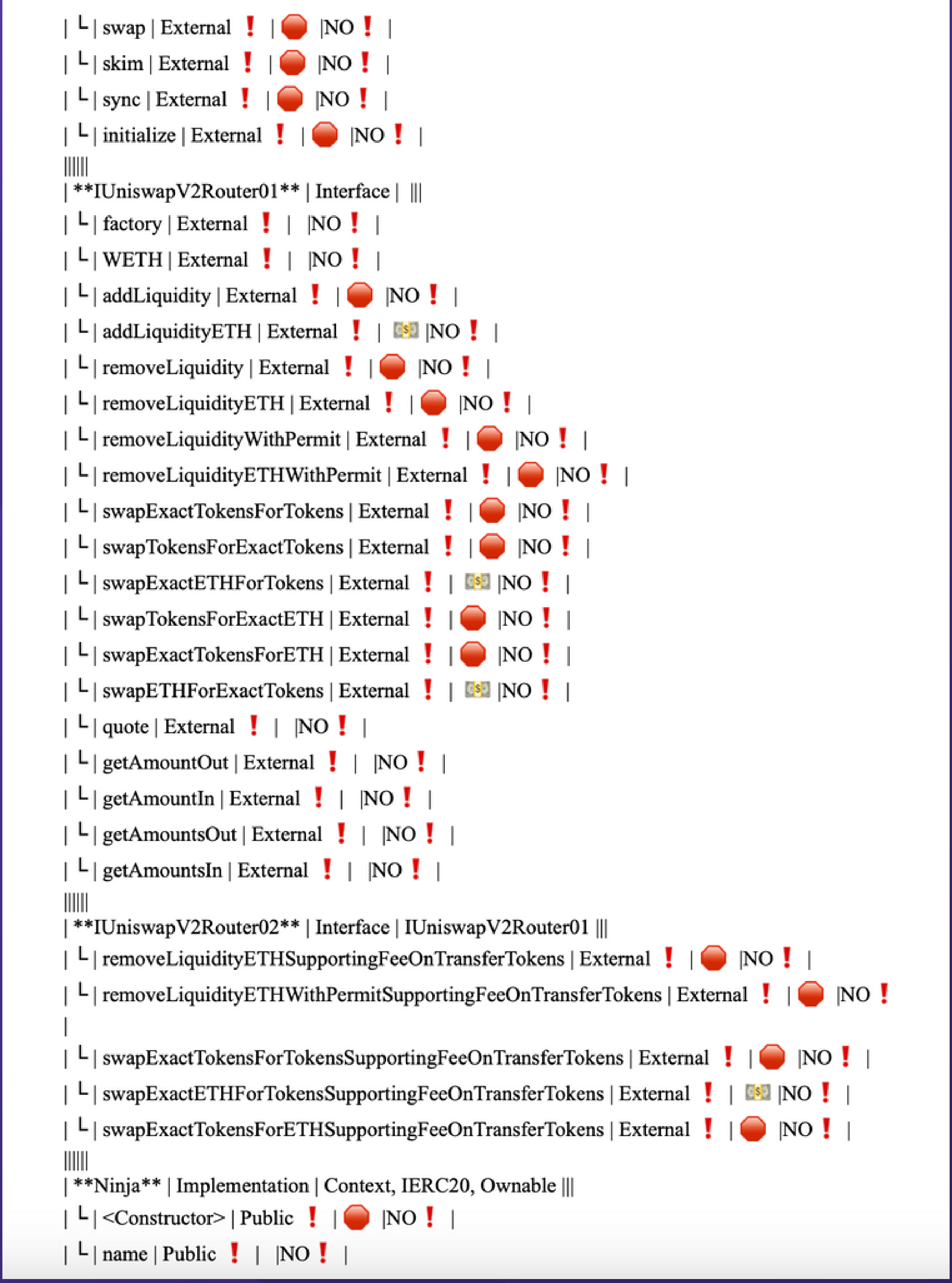
```
| Contract |       Type        |       Bases      |                  |                 |
|:----------:|:-------------------:|:----------------:|:----------------:|:---------------:|
|    └      |  **Function Name**  |  **Visibility**  |  **Mutability**  |  **Modifiers**  |
||||||
| **Context** | Implementation |  |||
| └ | _msgSender | Internal 🔒 |   | |
| └ | _msgData | Internal 🔒 |   | |
||||||
| **IERC20** | Interface |  |||
| └ | totalSupply | External ❗️ |   |NO❗️ |
| └ | balanceOf | External ❗️ |   |NO❗️ |
| └ | transfer | External ❗️ | 🛑  |NO❗️ |
| └ | allowance | External ❗️ |   |NO❗️ |
| └ | approve | External ❗️ | 🛑  |NO❗️ |
| └ | transferFrom | External ❗️ | 🛑  |NO❗️ |
||||||
| **SafeMath** | Library |  |||
| └ | add | Internal 🔒 |   | |
| └ | sub | Internal 🔒 |   | |
| └ | sub | Internal 🔒 |   | |
| └ | mul | Internal 🔒 |   | |
| └ | div | Internal 🔒 |   | |
| └ | div | Internal 🔒 |   | |
| └ | mod | Internal 🔒 |   | |
| └ | mod | Internal 🔒 |   | |
||||||
| **Address** | Library |  |||
| └ | isContract | Internal 🔒 |   | |
| └ | sendValue | Internal 🔒 | 🛑  | |
| └ | functionCall | Internal 🔒 | 🛑  | |
| └ | functionCall | Internal 🔒 | 🛑  | |
| └ | functionCallWithValue | Internal 🔒 | 🛑  | |
| └ | functionCallWithValue | Internal 🔒 | 🛑  | |
| └ | _functionCallWithValue | Private 🔐 | 🛑  | |
||||||
| **Ownable** | Implementation | Context |||
| └ | <Constructor> | Public ❗️ | 🛑  |NO❗️ |
| └ | owner | Public ❗️ |   |NO❗️ |
```

FUNCTION DETAILS



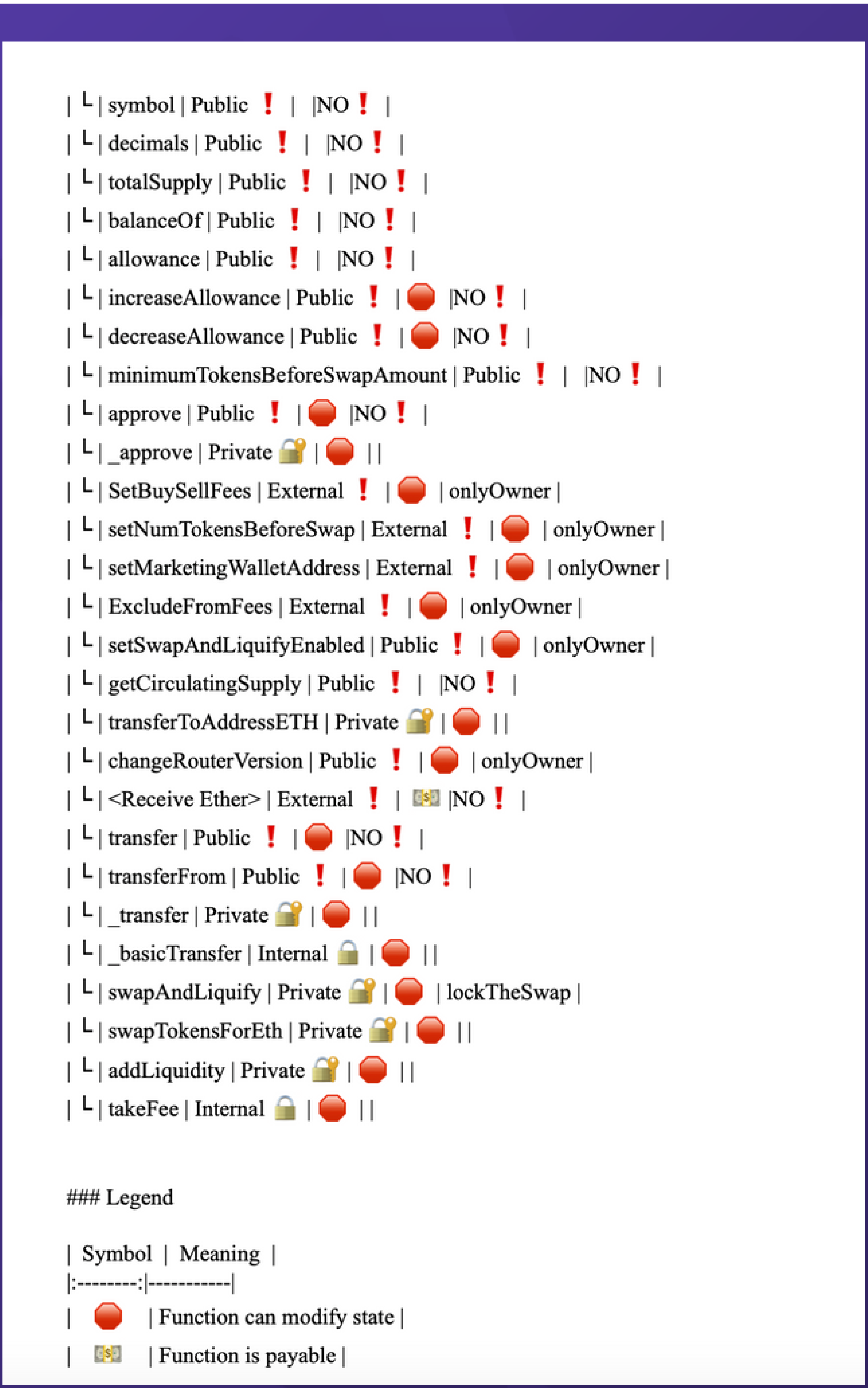
```
| └ | RenounceOwnership | Public ❗ | 🛑 | onlyOwner |
| └ | transferOwnership | Public ❗ | 🛑 | onlyOwner |
||||||
| **IUniswapV2Factory** | Interface | |||
| └ | feeTo | External ❗ | |NO ❗ |
| └ | feeToSetter | External ❗ | |NO ❗ |
| └ | getPair | External ❗ | |NO ❗ |
| └ | allPairs | External ❗ | |NO ❗ |
| └ | allPairsLength | External ❗ | |NO ❗ |
| └ | createPair | External ❗ | 🛑 |NO ❗ |
| └ | setFeeTo | External ❗ | 🛑 |NO ❗ |
| └ | setFeeToSetter | External ❗ | 🛑 |NO ❗ |
||||||
| **IUniswapV2Pair** | Interface | |||
| └ | name | External ❗ | |NO ❗ |
| └ | symbol | External ❗ | |NO ❗ |
| └ | decimals | External ❗ | |NO ❗ |
| └ | totalSupply | External ❗ | |NO ❗ |
| └ | balanceOf | External ❗ | |NO ❗ |
| └ | allowance | External ❗ | |NO ❗ |
| └ | approve | External ❗ | 🛑 |NO ❗ |
| └ | transfer | External ❗ | 🛑 |NO ❗ |
| └ | transferFrom | External ❗ | 🛑 |NO ❗ |
| └ | DOMAIN_SEPARATOR | External ❗ | |NO ❗ |
| └ | PERMIT_TYPEHASH | External ❗ | |NO ❗ |
| └ | nonces | External ❗ | |NO ❗ |
| └ | permit | External ❗ | 🛑 |NO ❗ |
| └ | MINIMUM_LIQUIDITY | External ❗ | |NO ❗ |
| └ | factory | External ❗ | |NO ❗ |
| └ | token0 | External ❗ | |NO ❗ |
| └ | token1 | External ❗ | |NO ❗ |
| └ | getReserves | External ❗ | |NO ❗ |
| └ | price0CumulativeLast | External ❗ | |NO ❗ |
| └ | price1CumulativeLast | External ❗ | |NO ❗ |
| └ | kLast | External ❗ | |NO ❗ |
| └ | burn | External ❗ | 🛑 |NO ❗ |
```

FUNCTION DETAILS



```
| └ | swap | External ❗ | 🛑 |NO ❗ |
| └ | skim | External ❗ | 🛑 |NO ❗ |
| └ | sync | External ❗ | 🛑 |NO ❗ |
| └ | initialize | External ❗ | 🛑 |NO ❗ |
||||||
| **IUniswapV2Router01** | Interface | |||
| └ | factory | External ❗ | |NO ❗ |
| └ | WETH | External ❗ | |NO ❗ |
| └ | addLiquidity | External ❗ | 🛑 |NO ❗ |
| └ | addLiquidityETH | External ❗ | 💵 |NO ❗ |
| └ | removeLiquidity | External ❗ | 🛑 |NO ❗ |
| └ | removeLiquidityETH | External ❗ | 🛑 |NO ❗ |
| └ | removeLiquidityWithPermit | External ❗ | 🛑 |NO ❗ |
| └ | removeLiquidityETHWithPermit | External ❗ | 🛑 |NO ❗ |
| └ | swapExactTokensForTokens | External ❗ | 🛑 |NO ❗ |
| └ | swapTokensForExactTokens | External ❗ | 🛑 |NO ❗ |
| └ | swapExactETHForTokens | External ❗ | 💵 |NO ❗ |
| └ | swapTokensForExactETH | External ❗ | 🛑 |NO ❗ |
| └ | swapExactTokensForETH | External ❗ | 🛑 |NO ❗ |
| └ | swapETHForExactTokens | External ❗ | 💵 |NO ❗ |
| └ | quote | External ❗ | |NO ❗ |
| └ | getAmountOut | External ❗ | |NO ❗ |
| └ | getAmountIn | External ❗ | |NO ❗ |
| └ | getAmountsOut | External ❗ | |NO ❗ |
| └ | getAmountsIn | External ❗ | |NO ❗ |
||||||
| **IUniswapV2Router02** | Interface | IUniswapV2Router01 |||
| └ | removeLiquidityETHSupportingFeeOnTransferTokens | External ❗ | 🛑 |NO ❗ |
| └ | removeLiquidityETHWithPermitSupportingFeeOnTransferTokens | External ❗ | 🛑 |NO ❗
|
| └ | swapExactTokensForTokensSupportingFeeOnTransferTokens | External ❗ | 🛑 |NO ❗ |
| └ | swapExactETHForTokensSupportingFeeOnTransferTokens | External ❗ | 💵 |NO ❗ |
| └ | swapExactTokensForETHSupportingFeeOnTransferTokens | External ❗ | 🛑 |NO ❗ |
||||||
| **Ninja** | Implementation | Context, IERC20, Ownable |||
| └ | <Constructor> | Public ❗ | 🛑 |NO ❗ |
| └ | name | Public ❗ | |NO ❗ |
```

FUNCTION DETAILS



```
| └ | symbol | Public ❗ | |NO ❗ |
| └ | decimals | Public ❗ | |NO ❗ |
| └ | totalSupply | Public ❗ | |NO ❗ |
| └ | balanceOf | Public ❗ | |NO ❗ |
| └ | allowance | Public ❗ | |NO ❗ |
| └ | increaseAllowance | Public ❗ | 🛑 |NO ❗ |
| └ | decreaseAllowance | Public ❗ | 🛑 |NO ❗ |
| └ | minimumTokensBeforeSwapAmount | Public ❗ | |NO ❗ |
| └ | approve | Public ❗ | 🛑 |NO ❗ |
| └ | _approve | Private 🔐 | 🛑 | |
| └ | SetBuySellFees | External ❗ | 🛑 | onlyOwner |
| └ | setNumTokensBeforeSwap | External ❗ | 🛑 | onlyOwner |
| └ | setMarketingWalletAddress | External ❗ | 🛑 | onlyOwner |
| └ | ExcludeFromFees | External ❗ | 🛑 | onlyOwner |
| └ | setSwapAndLiquifyEnabled | Public ❗ | 🛑 | onlyOwner |
| └ | getCirculatingSupply | Public ❗ | |NO ❗ |
| └ | transferToAddressETH | Private 🔐 | 🛑 | |
| └ | changeRouterVersion | Public ❗ | 🛑 | onlyOwner |
| └ | <Receive Ether> | External ❗ | 💵 |NO ❗ |
| └ | transfer | Public ❗ | 🛑 |NO ❗ |
| └ | transferFrom | Public ❗ | 🛑 |NO ❗ |
| └ | _transfer | Private 🔐 | 🛑 | |
| └ | _basicTransfer | Internal 🔒 | 🛑 | |
| └ | swapAndLiquify | Private 🔐 | 🛑 | lockTheSwap |
| └ | swapTokensForEth | Private 🔐 | 🛑 | |
| └ | addLiquidity | Private 🔐 | 🛑 | |
| └ | takeFee | Internal 🔒 | 🛑 | |


### Legend

| Symbol | Meaning |
|:--------:|-----------|
| 🛑 | Function can modify state |
| 💵 | Function is payable |
```

TESTNET VERSION

Adding Liquidity ✓
Tx:
https://testnet.bscscan.com/tx/0x734689cb2873bf94a946201b11de2d4f8b68d39dd2eab86a81a4409666632a57

===

Buying when excluded from fees ✓
Tx (0% tax):
https://testnet.bscscan.com/tx/0x1438bbb16dd82c56e37a031e18ee300851da2c957454be58f1ac842e23ffd33c

===

Selling when excluded from fees ✓
Tx (0% tax):
https://testnet.bscscan.com/tx/0xbe12031484b323e26ee44331f6d34e3772f9e5a2f4ab5d41be3dca666ba9fd23

===

Transferring when excluded from fees ✓
Tx (0% tax):
https://testnet.bscscan.com/tx/0x3c3d2390734fd225fa80ba52969171b816eae5c47b3e2b741fcedb95be07fcba

===

Buying ✓
Tx (0-10% tax):
https://testnet.bscscan.com/tx/0xe7ac762519c16833c56a3ad2ad77a883315720e8b265a9c253f52be830108abd

TESTNET VERSION

Selling ✓
Tx (0-10% tax):
https://testnet.bscscan.com/tx/0xc29e3fc8149065455aeb6728b91b032287b5989dceaa5c5f6206ef505c0c3ee5

==

Transferring ✓
Tx (0% tax):
https://testnet.bscscan.com/tx/0x3dac403665179c7f44d1aeee947d9dc409e5a9fc6a0141e45937631f825870f2

==

Internal swap (BNB to marketing wallet | reward token to dividend tracker | reward distribution) ✓
Tx:
https://testnet.bscscan.com/tx/0xc29e3fc8149065455aeb6728b91b032287b5989dceaa5c5f6206ef505c0c3ee5

MANUAL REVIEW

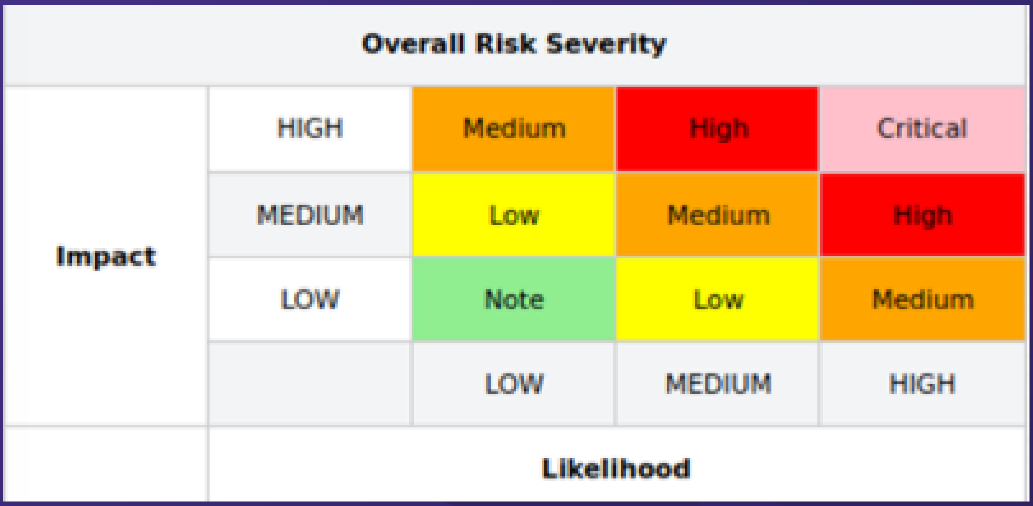
Severity Criteria

Expelee assesses the severity of disclosed vulnerabilities according to methodology based on OWASP standarts.

Vulnerabilities are dividend into three primary risk categroies:
High
Medium
Low

High-level considerations for vulnerabilities span the following key areas when conducting assessments:

- Malicious input handling
- Escalation of privileges
- Arithmetic
- Gas use



Overall Risk Severity				
Impact	HIGH	Medium	High	Critical
	MEDIUM	Low	Medium	High
	LOW	Note	Low	Medium
		LOW	MEDIUM	HIGH
	Likelihood			

HIGH RISK FINDING

Unbounded swap threshold

Category: Centralization
Status: Open
Impact: High

Overview:
The contract owner is able to set swap threshold to 0 which disables sell/transfers as contract tries to perform internal swap with 0 tokens.

```
 function setNumTokensBeforeSwap(uint256 newLimit) external onlyOwner {
   minimumTokensBeforeSwap = newLimit;
 }
```

Suggestion:
Ensure that minimumTokensBeforeSwap is always greater than zero.

```
 function setNumTokensBeforeSwap(uint256 newLimit) external onlyOwner {
   require(minimumTokensBeforeSwap > 10 ** decimal(), “swap threshold
must be greater than 1 token’);
   minimumTokensBeforeSwap = newLimit;
 }
```

HIGH RISK FINDING

Changing router

Category: Logical
Status: Open
Impact: High

Overview:
Owner is able to update swap router that is used for performing internal swap. Setting router to a malicious contract could revert internal swaps and eventually whole transfer/sell transaction.

```
 function changeRouterVersion(address newRouterAddress) public onlyOwner
returns (address newPairAddress) {
   IUniswapV2Router02 _uniswapV2Router =
IUniswapV2Router02(newRouterAddress);
   newPairAddress =
IUniswapV2Factory(_uniswapV2Router.factory()).getPair(address(this),
_uniswapV2Router.WETH());
   if (
     newPairAddress == address(0) //Create If Doesnt exist
   ) {
     newPairAddress =
       IUniswapV2Factory(_uniswapV2Router.factory()).createPair(address(this),
_uniswapV2Router.WETH());
   }
   uniswapPair = newPairAddress; //Set new pair address
   uniswapV2Router = _uniswapV2Router; //Set new router address
   isMarketPair[address(uniswapPair)] = true;
 }
```

Suggestion:
Ensure that router is immutable in order to mitigate this logical issue.

MEDIUM RISK FINDING

Owner receiving LP shares

Category: **Centralization**
Status: Open
Impact: **Medium**

Overview:
After each auto-liquidity (internal swap), owner receives the minted LP tokens. This accumulated LP tokens can be used to remove a portion of liquidity pool. The impact could be little to high depending on this LP tokens and total LP tokens which were initialiy minted

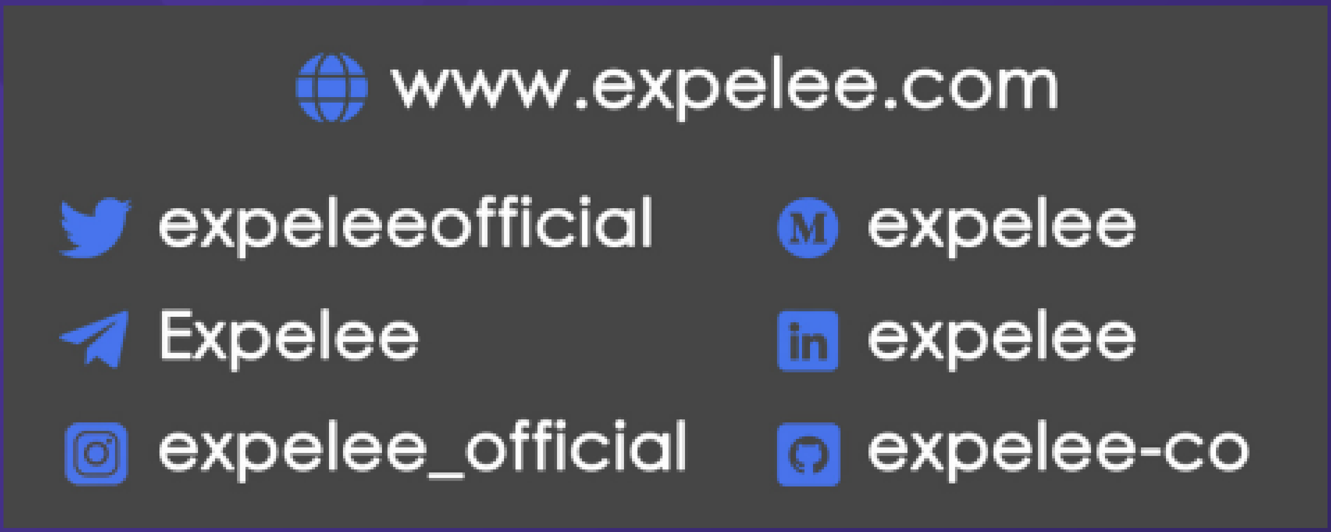
```
function addLiquidity(uint256 tokenAmount, uint256 ethAmount) private {
  // approve token transfer to cover all possible scenarios
  _approve(address(this), address(uniswapV2Router), tokenAmount);

  // add the liquidity
  uniswapV2Router.addLiquidityETH{value: ethAmount}(
    address(this),
    tokenAmount,
    0, // slippage is unavoidable
    0, // slippage is unavoidable
    owner(),
    block.timestamp
  );
}
```

Suggestion:
Its suggested to burn or Lock new LP tokens.

ABOUT EXPELEE

Expelee is a product-based aspirational Web3 start-up. Coping up with numerous solutions for blockchain security and constructing a Web3 ecosystem from deal making platform to developer hosting open platform, while also developing our own commercial and sustainable blockchain.



www.expelee.com

expeleeofficial	expelee
Expelee	expelee
expelee_official	expelee-co

expelee

Building the Futuristic Blockchain Ecosystem

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. Team provides no guarantess against the sale of team tokens or the removal of liquidity by the project audited in this document.

Always do your own research and project yourselves from being scammed. The Expelee 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 Expelee receive a payment to manipulate those results or change the awarding badge that we will be adding in our website. Alway 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 Expelee team disclaims any liability for the resulting losses.

expelee

Building the Futuristic Blockchain Ecosystem