auditace.tech

AuditAce
FROM INCEPTION TO SUCCESS

# Smart Contract Audit

FOR

# SHIB VS PEPE

DATED : 14 September 23'

# AUDIT SUMMARY

**Project name** – SHIB VS PEPE

**Date**: 14 September 2023

**Scope of Audit-** Audit Ace was consulted to conduct the smart contract audit of the solidity source codes.

**Audit Status: Passed**

## Issues Found

| Status | Critical | High | Medium | Low | Suggestion |
|---|---|---|---|---|---|
| Open | 0 | 0 | 0 | 0 | 0 |
| Acknowledged | 0 | 0 | 0 | 0 | 0 |
| Not Resolved | 0 | 0 | 0 | 0 | 1 |

# USED TOOLS

## Tools:

**1- Manual Review:**
A line by line code review has been performed by audit ace team.

**2- BSC Test Network:** All tests were conducted on the BSC Test network, and each test has a corresponding transaction attached to it. These tests can be found in the "Functional Tests" section of the report.

**3- Slither :**
The code has undergone static analysis using Slither.

**Testnet version:**
The tests were performed using the contract deployed on the BSC Testnet, which can be found at the following address:
https://testnet.bscscan.com/token/0xbf22Ed163Ff4bceab0be42D87E8F582ddce83812

# Token Information

**Token Address :**
0x9653304f08B81F12627525E9A952e5fC30fA9758

**Name:** SHIB VS PEPE🔥🔥🔥

**Symbol:** SVSP

**Decimals:** 18

**Netowrk:** Binance smart chain

**Token Type:** BEP20

**Owner:** 0xEF3e40853AE4268352996486878516aA97841d88
(at time of writing the audit)

**Deployer:**
0xEF3e40853AE4268352996486878516aA97841d88

**Token Supply:** 1,000,000,000,000,000

**Checksum:**
0d13ff50475c3fea38371e558f4b13bc5a383542

**Testnet version:**
The tests were performed using the contract deployed on the BSC Testnet, which can be found at the following address:
https://testnet.bscscan.com/token/0xbf22Ed163Ff4bceab0be42D87E8F582ddce83812

# TOKEN OVERVIEW

**buy fee:** 0-25%

**Sell fee:** 0-25%

**transfer fee:** 0-25%

**Fee Privilege:** Owner

**Ownership:** Owned

**Minting:** None

**Max Tx:** None

**Blacklist:** No

**Other Privileges:**

- Initial distribution of the tokens
- Modifying fees

# AUDIT METHODOLOGY

The auditing process will follow a routine as special considerations by Auditace:

- Review of the specifications, sources, and instructions provided to Auditace to make sure the contract logic meets the intentions of the client without exposing the user's funds to risk.
- Manual review of the entire codebase by our experts, which is the process of reading source code line-by-line in an attempt to identify potential vulnerabilities.
- Specification comparison is the process of checking whether the code does what the specifications, sources, and instructions provided to Auditace describe.
- Test coverage analysis determines whether the test cases are covering the code and how much code isexercised when we run the test cases.
- Symbolic execution is analysing a program to determine what inputs cause each part of a program to execute.
- Reviewing the codebase to improve maintainability, security, and control based on the established industry and academic practices.

# VULNERABILITY CHECKLIST

- [x] Return values of low-level calls
- [x] Private modifier
- [x] Multiple Sends
- [x] Using Suicide
- [x] Gas Limitand Loops
- [x] Address hardcoded
- [x] Exception Disorder
- [x] Using inline assembly
- [x] Divide before multiply
- [x] Missing Zero Address Validation
- [x] Compiler version not fixed
- [x] **Gasless Send**
- [x] Using block.timestamp
- [x] Re-entrancy
- [x] Tautology or contradiction
- [x] Timestamp Dependence
- [x] Revert/require functions
- [x] Use of tx.origin
- [x] Integer overflow/underflow
- [x] Dangerous strict equalities
- [x] Using SHA3
- [x] Using throw

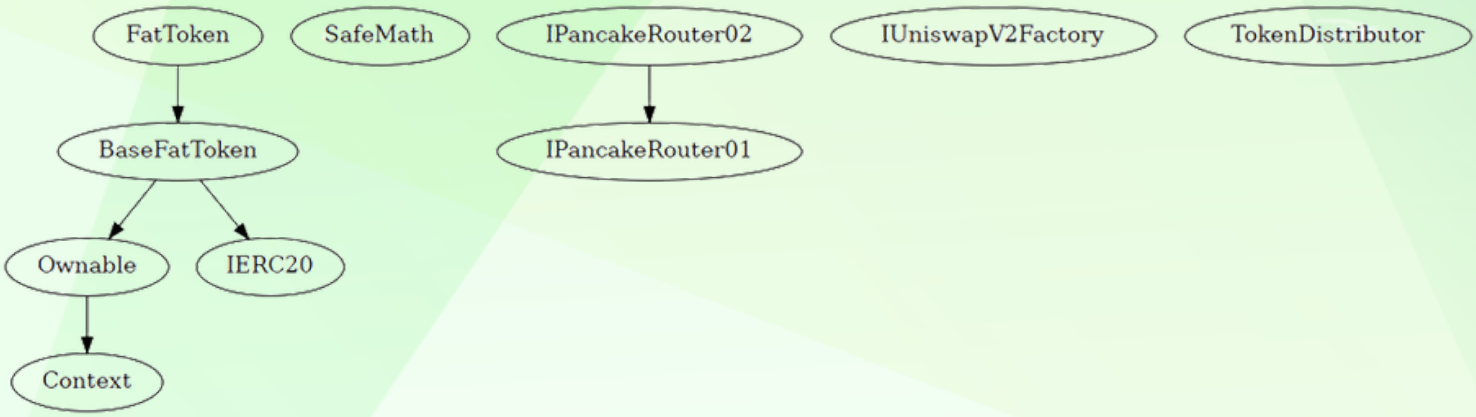
# CLASSIFICATION OF RISK

| Severity | Description |
| --- | --- |
| Critical | These vulnerabilities could be exploited easily and can lead to asset loss, data loss, asset, or data manipulation. They should be fixed right away. |
| High-Risk | A vulnerability that affects the desired outcome when using a contract, or provides the opportunity to use a contract in an unintended way. |
| Medium-Risk | A vulnerability that could affect the desired outcome of executing the contract in a specific scenario. |
| Low-Risk | A vulnerability that does not have a significant impact on possible scenarios for the use of the contract and is probably subjective. |
| Gas Optimization /Suggestion | A vulnerability that has an informational character but is not affecting any of the code. |

# Findings

| Severity | Found |
| --- | --- |
| Critical | 0 |
| High-Risk | 0 |
| Medium-Risk | 0 |
| Low-Risk | 0 |
| Gas Optimization / Suggestions | 1 |

# INHERITANCE TREE



# POINTS TO NOTE

- Owner is able to change current fees within 0-25% for buy/sell/transfers
- Owner is not able to blacklist an arbitrary address.
- Owner is not able to mint new tokens
- Owner is not able to disable trades
- Owner is not able to set maximum buy/sell/transfer amounts

# STATIC ANALYSIS

```
Reentrancy in FatToken.swapTokenForFund(uint256,uint256) (contracts/Token.sol#772-877):
        External calls:
        - fundAddress.transfer(fundAmount) (contracts/Token.sol#821)
        External calls sending eth:
        - fundAddress.transfer(fundAmount) (contracts/Token.sol#821)
        - _swapRouter.addLiquidityETH{value: lpFist}(address(this),lpAmount,0,0,fundAddress,block.timestamp) (contracts/Token.sol#825-836)
        Event emitted after the call(s):
        - Failed_AddLiquidityETH() (contracts/Token.sol#835)
Reentrancy in FatToken.transferFrom(address,address,uint256) (contracts/Token.sol#528-540):
        External calls:
        - _transfer(sender,recipient,amount) (contracts/Token.sol#533)
                - fundAddress.transfer(fundAmount) (contracts/Token.sol#821)
        External calls sending eth:
        - _transfer(sender,recipient,amount) (contracts/Token.sol#533)
                - fundAddress.transfer(fundAmount) (contracts/Token.sol#821)
                - _swapRouter.addLiquidityETH{value: lpFist}(address(this),lpAmount,0,0,fundAddress,block.timestamp) (contracts/Token.sol#825-836)
        State variables written after the call(s):
        - _allowances[sender][msg.sender] = _allowances[sender][msg.sender] - amount (contracts/Token.sol#535-537)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#reentrancy-vulnerabilities-4
INFO:Detectors:
Variable IPancakeRouter01.addLiquidity(address,address,uint256,uint256,uint256,uint256,address,uint256).amountADesired (contracts/Token.sol#193) is too similar to IPancakeRouter01.addLiquidit
y(address,address,uint256,uint256,uint256,uint256,address,uint256).amountBDesired (contracts/Token.sol#194)
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#variable-names-too-similar
INFO:Detectors:
BaseFatToken.deadAddress (contracts/Token.sol#297) should be constant
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant
INFO:Detectors:
BaseFatToken._mainPair (contracts/Token.sol#308) should be immutable
BaseFatToken._swapRouter (contracts/Token.sol#304) should be immutable
BaseFatToken.currency (contracts/Token.sol#275) should be immutable
BaseFatToken.currencyIsEth (contracts/Token.sol#264) should be immutable
BaseFatToken.decimals (contracts/Token.sol#294) should be immutable
BaseFatToken.enableKillBlock (contracts/Token.sol#267) should be immutable
BaseFatToken.enableOffTrade (contracts/Token.sol#266) should be immutable
BaseFatToken.enableRewardList (contracts/Token.sol#268) should be immutable
BaseFatToken.totalSupply (contracts/Token.sol#295) should be immutable
FatToken._tokenDistributor (contracts/Token.sol#432) should be immutable
FatToken.enableTransferFee (contracts/Token.sol#592) should be immutable
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-immutable
INFO:Slither:./contracts/Token.sol analyzed (10 contracts with 88 detectors), 83 result(s) found
```

**Result => A static analysis of contract's source code has been performed using slither,**

**No major issues were found in the output**

# CONTRACT ASSESMENT

| Contract | Type | Bases | | |
|:---|:---|:---|:---|:---|
| └ | **Function Name** | **Visibility** | **Mutability** | **Modifiers** |
| | | | | |
| **Context** | Implementation | | | |
| └ | _msgSender | Internal 🔒 | | |
| └ | _msgData | Internal 🔒 | | |
| | | | | |
| **Ownable** | Implementation | Context | | |
| └ | <Constructor> | Public ❗ | 🛑 | NO ❗ |
| └ | renounceOwnership | Public ❗ | 🛑 | onlyOwner |
| └ | transferOwnership | Public ❗ | 🛑 | onlyOwner |
| └ | owner | Public ❗ | | NO ❗ |
| | | | | |
| **SafeMath** | Library | | | |
| └ | add | Internal 🔒 | | |
| └ | sub | Internal 🔒 | | |
| └ | sub | Internal 🔒 | | |
| └ | mul | Internal 🔒 | | |
| └ | div | Internal 🔒 | | |
| └ | div | Internal 🔒 | | |
| └ | mod | Internal 🔒 | | |
| └ | mod | Internal 🔒 | | |
| | | | | |
| **IERC20** | Interface | | | |
| └ | name | External ❗ | | NO ❗ |
| └ | symbol | External ❗ | | NO ❗ |
| └ | totalSupply | External ❗ | | NO ❗ |
| └ | decimals | External ❗ | | NO ❗ |
| └ | balanceOf | External ❗ | | NO ❗ |
| └ | transfer | External ❗ | 🛑 | NO ❗ |
| └ | allowance | External ❗ | | NO ❗ |
| └ | approve | External ❗ | 🛑 | NO ❗ |
| └ | transferFrom | External ❗ | 🛑 | NO ❗ |

# CONTRACT ASSESMENT

| **IPancakeRouter01** | Interface | |||
| └ | factory | External ❗ | |NO❗ |
| └ | WETH | External ❗ | |NO❗ |
| └ | addLiquidity | External ❗ | 🔴 |NO❗ |
| └ | addLiquidityETH | External ❗ | 💵 |NO❗ |
||||||
| **IPancakeRouter02** | Interface | IPancakeRouter01 |||
| └ | swapExactTokensForTokensSupportingFeeOnTransferTokens | External ❗ | 🔴 |NO❗ |
| └ | swapExactTokensForETHSupportingFeeOnTransferTokens | External ❗ | 🔴 |NO❗ |
||||||
| **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❗ |
||||||
| **BaseFatToken** | Implementation | IERC20, Ownable |||
| └ | changeSwapLimit | External ❗ | 🔴 | onlyOwner |
| └ | changeWalletLimit | External ❗ | 🔴 | onlyOwner |
| └ | launch | External ❗ | 🔴 | onlyOwner |
| └ | disableSwapLimit | Public ❗ | 🔴 | onlyOwner |
| └ | disableWalletLimit | Public ❗ | 🔴 | onlyOwner |
| └ | disableChangeTax | Public ❗ | 🔴 | onlyOwner |
| └ | completeCustoms | External ❗ | 🔴 | onlyOwner |
| └ | transfer | External ❗ | 🔴 |NO❗ |
| └ | transferFrom | External ❗ | 🔴 |NO❗ |
| └ | setAntiSYNCEnable | Public ❗ | 🔴 | onlyOwner |
| └ | balanceOf | Public ❗ | |NO❗ |
| └ | allowance | Public ❗ | |NO❗ |
| └ | approve | Public ❗ | 🔴 |NO❗ |
| └ | _approve | Private 🔐 | 🔴 | |
| └ | setFeeWhiteList | External ❗ | 🔴 | onlyOwner |
| └ | multi_bclist | Public ❗ | 🔴 | onlyOwner |
||||||
| **TokenDistributor** | Implementation | |||
| └ | <Constructor> | Public ❗ | 🔴 |NO❗ |
||||||
| **FatToken** | Implementation | BaseFatToken |||
| └ | <Constructor> | Public ❗ | 🔴 |NO❗ |
| └ | transfer | Public ❗ | 🔴 |NO❗ |
| └ | transferFrom | Public ❗ | 🔴 |NO❗ |

# CONTRACT ASSESMENT

| └ | setFundAddress | External ❗ | 🛑 | onlyOwner |
| └ | isContract | Private 🔐 | | |
| └ | setkb | Public ❗ | 🛑 | onlyOwner |
| └ | isReward | Public ❗ | |NO❗ |
| └ | setAirDropEnable | Public ❗ | 🛑 | onlyOwner |
| └ | _basicTransfer | Internal 🔒 | 🛑 | |
| └ | setAirdropNumbs | Public ❗ | 🛑 | onlyOwner |
| └ | setEnableTransferFee | Public ❗ | 🛑 | onlyOwner |
| └ | _transfer | Private 🔐 | 🛑 | |
| └ | setTransferFee | Public ❗ | 🛑 | onlyOwner |
| └ | _tokenTransfer | Private 🔐 | 🛑 | |
| └ | swapTokenForFund | Private 🔐 | 🛑 | lockTheSwap |
| └ | _takeTransfer | Private 🔐 | 🛑 | |
| └ | setSwapPairList | External ❗ | 🛑 | onlyOwner |
| └ | <Receive Ether> | External ❗ | 💵 |NO❗ |

### Legend

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

# FUNCTIONAL TESTING

1- Adding liquidity (passed):
https://testnet.bscscan.com/tx/0xdfe0336693d2a6e39c94f2bf8bd39143e90d25341a1448ab9921fac98676a0b1

2- Buying when excluded from fees (0% tax) (passed):
https://testnet.bscscan.com/tx/0x0aa1713014353e46910d725938cf46e184ff062178a33c07b64901d636f5ec35

3- Selling when excluded from fees (0% tax) (passed):
https://testnet.bscscan.com/tx/0x900ccc6bfc52c79584be530b810832c564c114de75f117205134042647273233

4- Transferring when excluded from fees (0% tax) (passed):
https://testnet.bscscan.com/tx/0x30d7cd6c0a6b8fb381998a568040fa7be07d6516de21e85f298b2be643cd5de4

5- Buying when not excluded from fees (0-25% tax) (passed):
https://testnet.bscscan.com/tx/0x46a3aa66b44374c1b1bd173b9917d99e01f8e0fb7052696cee6aa1561dc4b059

6- Selling when not excluded from fees (0-25% tax) (passed):
https://testnet.bscscan.com/tx/0x4738f4a5cac70a2f2b80c4f610f5da6bab1a8203a4fdcb2479fb56c816e281a9

7- Transferring when not excluded from fees (0% tax) (passed):
https://testnet.bscscan.com/tx/0xa319e3c05f7043865ca5bfdf1078f1932ecaa901f2434b7f7cc71cb015b6676b

8- Internal swap (passed):
https://testnet.bscscan.com/tx/0xf0d9cf0999ab080b2099593243f0a2b3edd76969d96ae059364e638af266e5b8

# MANUAL TESTING

**Centralization – Maximum buy and sell**
**Severity: Informational**
**function:** launch
**Status: Not Resolved**
**Overview:**
Owner is able to set maximum buy or sell amount each to zero.

```
  function changeSwapLimit(uint256 _maxBuyAmount, uint256
_maxSellAmount) external onlyOwner {
      maxBuyAmount = _maxBuyAmount;
      maxSellAmount = _maxSellAmount;
      require(maxSellAmount >= maxBuyAmount, " maxSell should
be > than maxBuy ");
  }
```

**Suggestion**
Set an upper bound for maximum amount of buy and sell limits.

```
  function changeSwapLimit(uint256 _maxBuyAmount, uint256
_maxSellAmount) external onlyOwner {
      maxBuyAmount = _maxBuyAmount;
      maxSellAmount = _maxSellAmount;
      reuqire(maxBuyAmount >= totalSupply() / 1000, “Can’t set
mximum buy lower than 0.1% of supply”);
      reuqire(maxSellAmount >= totalSupply() / 1000, “Can’t set
mximum sell lower than 0.1% of supply”));
      require(maxSellAmount >= maxBuyAmount, " maxSell should
be > than maxBuy ");
  }
```

# 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 guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed. The Auditace 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 Auditace receive a payment to manipulate those results or change the awarding badge that we will be adding in our website. Always Do your own research and protect yourselves from scams. This document should not be presented as a reason to buy or not buy any particular token. The Auditace team disclaims any liability for the resulting losses.

# ABOUT AUDITACE

We specializes in providing thorough and reliable audits for Web3 projects. With a team of experienced professionals, we use cutting-edge technology and rigorous methodologies to evaluate the security and integrity of blockchain systems. We are committed to helping our clients ensure the safety and transparency of their digital assets and transactions.


**https://auditace.tech/**

**https://t.me/Audit_Ace**

**https://twitter.com/auditace_**

**https://github.com/Audit-Ace**