



# Smart Contract Audit

FOR

# Bankp2pGold

DATED : 7 June 23'



# High Risk Issues

## Centralization – Excessive Fees

**Severity:** High

**function:** setAllFeePercent

**Status:** Open

### Overview:

Owner is able to set buy/sell/transfer fees up to 60%

```
function setAllFeePercent(
    uint8 taxFee,
    uint8 liquidityFee,
    uint8 burnFee,
    uint8 marketingFee,
    uint8 devFee,
    uint8 rewardFee,
    uint8 extraSellFee
) external onlyOwner {
    uint8 _maxFee = 10;
    require(taxFee >= 0 && taxFee <= maxTaxFee, "TF err");
    require(liquidityFee >= 0 && liquidityFee <= maxLiqFee, "LF err");
    require(burnFee >= 0 && burnFee <= maxBurnFee, "BF err");
    require(extraSellFee >= 0 && extraSellFee <= maxExtraSellFee, "ESF err");
    require(marketingFee >= 0 && marketingFee <= _maxFee, "WF err");
    require(devFee >= 0 && devFee <= _maxFee, "WFT err");
    require(rewardFee >= 0 && rewardFee <= _maxFee, "RF err");
    //both tax fee and reward fee cannot be set
    require(rewardFee == 0 || taxFee == 0, "RT fee err");
    _walletFee = marketingFee;
    _walletDevFee = devFee;
    _rewardFee = rewardFee;
    _taxFee = taxFee;
    _liquidityFee = liquidityFee;
    _burnFee = burnFee;
    _extraSellFee = extraSellFee;
}
```

### Suggestion

According to pinksale safu criteria, its suggested to have a limitation of max 10% for buy/sell/transfer fees (each one seperately)

0 <= Buy Fees <= 10

0 <= Sell Fees <= 10

0 <= Transfer Fees <= 10



# High Risk Issues

## Configuration – Invalid reward tokens

Severity: **High**

**function:** setMarketingFeeToken - setDevFeeToken

**Status:** Open

### Overview:

Owner is able to change marketing and dev reward tokens to any arbitrary address. Setting these addresses to an EOA, or an ERC20 token without liquidity on PCS V2 can disable sells.

```
function setMarketingFeeToken(address feeToken) external onlyOwner {  
    marketingFeeToken = feeToken;  
}
```

```
function setDevFeeToken(address feeToken) external onlyOwner {  
    devFeeToken = feeToken;  
}
```

### Suggestion

Ensure that dev and marketing fee tokens are immutable



# AUDIT SUMMARY

**Project name - Bankp2pGold**

**Date:** 7 June, 2023

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

**Audit Status: Passed With High Risk**

## Issues Found

Status	Critical	High	Medium	Low	Suggestion
Open	0	2	1	0	0
Acknowledged	0	0	0	0	0
Resolved	0	0	0	0	0



# 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 done on BSC Test network, each test has its transaction has attached to it.

### 3- Slither : Static Analysis

**Testnet Link:** all tests were done using this contract, tests are done on BSC Testnet

<https://testnet.bscscan.com/token/0x68C017806319B8214835AB8FcaDE455e65078178>

---



# Token Information

---

**Token Name :** Bankp2pGold

**Token Symbol:** Bankp2pGold

**Decimals:** 18

**Token Supply:** 100,000,000

**Token Address:**

0x1E3B32643666565a787b312cBDFDac16B19EA161

**Checksum:**

0d0e6bc06f515a1ad622d942af16c4611db4d8b9

**Owner:** -

0x41f38Cc1Fb5D9E3b3D535827AD1C77f586F17796

**Deployer:**

0x41f38Cc1Fb5D9E3b3D535827AD1C77f586F17796

---



# TOKEN OVERVIEW

---

**Fees:**

Buy Fees: 0-60%

Sell Fees: 0-60%

Transfer Fees: 0-60%

---

**Fees Privilege:** Owner

---

**Ownership :** Owned

---

**Minting:** No mint function

---

**Max Tx Amount/ Max Wallet Amount:** none

---

**Blacklist:** No

---

**Other Privileges:** - changing fees

- initial distribution of the tokens

---

---



# 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 is exercised 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



Return values of low-level calls



**Gasless Send**



Private modifier



Using block.timestamp



Multiple Sends



Re-entrancy



Using Suicide



Tautology or contradiction



Gas Limit and Loops



Timestamp Dependence



Address hardcoded



Revert/require functions



Exception Disorder



Use of tx.origin



Using inline assembly



Integer overflow/underflow



Divide before multiply



Dangerous strict equalities



Missing Zero Address Validation



Using SHA3



Compiler version not fixed



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	2
◆ Medium-Risk	1
◆ Low-Risk	0
◆ Gas Optimization / Suggestions	0



# CONTRACT ASSESSMENT

Contract	Type	Bases			
	L	**Function Name**	**Visibility**	**Mutability**	**Modifiers**
**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 !					
**SafeMath**   Library					
L   add   Internal					
L   sub   Internal					
L   sub   Internal					
L   mul   Internal					
L   div   Internal					
L   div   Internal					
L   mod   Internal					
L   mod   Internal					
**Context**   Implementation					
L   _msgSender   Internal					
L   _msgData   Internal					
**SafeMathInt**   Library					
L   mul   Internal					
L   div   Internal					
L   sub   Internal					
L   add   Internal					
L   abs   Internal					
L   toUint256Safe   Internal					
**SafeMathUint**   Library					
L   toInt256Safe   Internal					
**IterableMapping**   Library					
L   get   Internal					
L   getIndexOfKey   Internal					
L   getKeyAtIndex   Internal					
L   size   Internal					



# CONTRACT ASSESSMENT

```
| L | set | Internal 🔒 | ● | | | |
| L | remove | Internal 🔒 | ● | |
|||||||
| **Address** | Library | |||
| L | isContract | Internal 🔒 | |||
| L | sendValue | Internal 🔒 | ● | |
| L | functionCall | Internal 🔒 | ● | |
| L | functionCall | Internal 🔒 | ● | |
| L | functionCallWithValue | Internal 🔒 | ● | |
| L | functionCallWithValue | Internal 🔒 | ● | |
| L | _functionCallWithValue | Private 🔒 | ● | |
|||||||
| **SafeERC20** | Library | |||
| L | safeTransfer | Internal 🔒 | ● | |
| L | safeTransferFrom | Internal 🔒 | ● | |
| L | safeApprove | Internal 🔒 | ● | |
| L | safeIncreaseAllowance | Internal 🔒 | ● | |
| L | safeDecreaseAllowance | Internal 🔒 | ● | |
| L | _callOptionalReturn | Private 🔒 | ● | |
|||||||
| **Ownable** | Implementation | Context |||
| L | <Constructor> | Public ! | ● | NO ! | |
| L | owner | Public ! | | NO ! | |
| L | renounceOwnership | Public ! | ● | onlyOwner | |
| L | transferOwnership | Public ! | ● | onlyOwner | |
|||||||
| **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 ! |
```



# CONTRACT ASSESSMENT

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 !	
\*\*ProToken\*\*	Implementation	Context, IERC20, Ownable			
L	<Constructor>	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	decreaseAllowance	Public !	●	NO !	
L	totalFees	Public !		NO !	
L	deliver	Public !	●	NO !	
L	reflectionFromToken	Public !		NO !	
L	tokenFromReflection	Public !		NO !	
L	excludeFromFee	Public !	●	onlyOwner	
L	setAllFeePercent	External !	●	onlyOwner	



# CONTRACT ASSESSMENT

```
| L | setSwapAndLiquifyEnabled | Public ! | ○ | onlyOwner |
| L | setSwapAmount | External ! | ○ | onlyOwner |
| L | <Receive Ether> | External ! | SD | NO !
| L | _reflectFee | Private 🔒 | ○ || |
| L | _getValues | Private 🔒 | || |
| L | _getTValues | Private 🔒 | || |
| L | _getRValues | Private 🔒 | || |
| L | _getRate | Private 🔒 | || |
| L | _getCurrentSupply | Private 🔒 | || |
| L | _takeLiquidity | Private 🔒 | ○ || |
| L | calculateTaxFee | Private 🔒 | || |
| L | calculateLiquidityFee | Private 🔒 | || |
| L | removeAllFee | Private 🔒 | ○ || |
| L | restoreAllFee | Private 🔒 | ○ || |
| L | isExcludedFromFee | Public ! | | NO !
| L | _approve | Private 🔒 | ○ || |
| L | _transfer | Private 🔒 | ○ || |
| L | swapAndLiquify | Private 🔒 | ○ | lockTheSwap |
| L | swapTokensForBNB | Private 🔒 | ○ || |
| L | swapBNBForTokens | Private 🔒 | ○ || |
| L | swapTokensForFeeToken | Private 🔒 | ○ || |
| L | addLiquidity | Private 🔒 | ○ || |
| L | _tokenTransfer | Private 🔒 | ○ || |
| L | _transferStandard | Private 🔒 | ○ || |
| L | _transferToExcluded | Private 🔒 | ○ || |
| L | _transferFromExcluded | Private 🔒 | ○ || |
| L | _transferBothExcluded | Private 🔒 | ○ || |
| L | _tokenTransferNoFee | Private 🔒 | ○ || |
| L | transferEth | Private 🔒 | ○ || |
| L | recoverFunds | External ! | ○ | onlyOwner |
| L | recoverBEP20 | External ! | ○ | onlyOwner |
| L | sendTaxes | Internal 🔒 | ○ || |
| L | process | Public ! | ○ | NO !
| L | processAccount | Internal 🔒 | ○ || |
| L | excludeFromDividends | Public ! | ○ | onlyOwner |
| L | canAutoClaim | Private 🔒 | || |
| L | dividendOf | Public ! | | NO !
| L | withdrawableDividendOf | Public ! | | NO !
| L | withdrawnDividendOf | Public ! | | NO !
| L | accumulativeDividendOf | Public ! | | NO !
| L | _withdrawDividendOfUser | Internal 🔒 | ○ ||
```



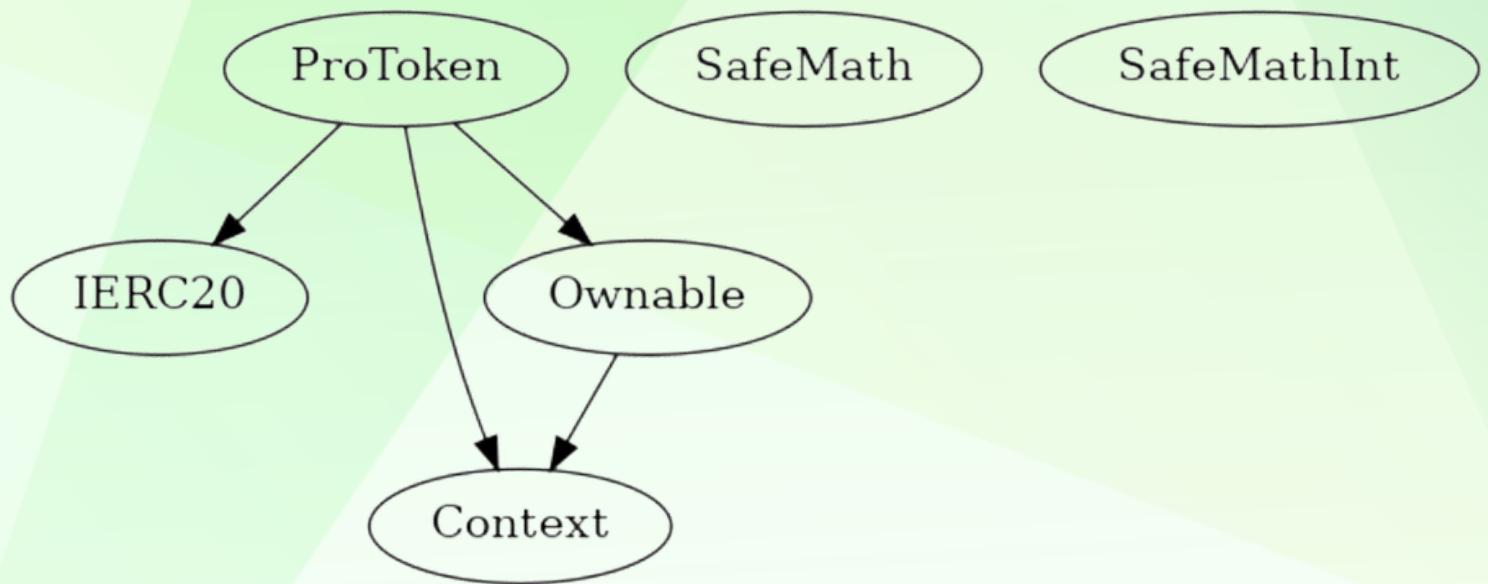
# CONTRACT ASSESSMENT

```
| L | withdrawDividend | Public ! | [ ] | NO ! |
| L | setMinimumTokenBalanceForDividends | External ! | [ ] | onlyOwner |
| L | excludeFromReward | Public ! | [ ] | onlyOwner |
| L | includeInReward | External ! | [ ] | onlyOwner |
| L | isExcludedFromReward | Public ! | [ ] | NO !
| L | getNumberOfDividendTokenHolders | External ! | [ ] | NO !
| L | processDividendTracker | External ! | [ ] | NO !
| L | claim | External ! | [ ] | NO !
| L | distributeDividends | Internal [ ] | [ ] ||
| L | _dtransfer | Internal [ ] | [ ] ||
| L | _dmint | Internal [ ] | [ ] ||
| L | _dburn | Internal [ ] | [ ] ||
| L | _setBalance | Internal [ ] | [ ] ||
| L | setBalance | Private [ ] | [ ] ||
| L | setFeeWallet | External ! | [ ] | onlyOwner |
| L | setMarketingFeeToken | External ! | [ ] | onlyOwner |
| L | setDevWallet | External ! | [ ] | onlyOwner |
| L | setDevFeeToken | External ! | [ ] | onlyOwner |
```

### Legend

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

# INHERITANCE TREE





## POINTS TO NOTE

---

- owner is able to change buy/sell/transfer taxes in range 0-60%
- owner is not able to blacklist an arbitrary wallet
- owner is not able to set limit for buy/sell/transfer/holding amounts
- owner is not able to mint new tokens
- owner is not able to disable trades



# STATIC ANALYSIS

```
ProToken.slitherConstructorVariables() (contracts/Token.sol#872-2036) uses literals with too many digits:  
    - gasForProcessing = 300000 (contracts/Token.sol#913)  
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#too-many-digits  
  
SafeMathInt.MAX_INT256 (contracts/Token.sol#251) is never used in SafeMathInt (contracts/Token.sol#249-306)  
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#unused-state-variable  
  
ProToken.claimWait (contracts/Token.sol#909) should be constant  
ProToken.dead (contracts/Token.sol#880) should be constant  
ProToken.gasForProcessing (contracts/Token.sol#913) should be constant  
ProToken.isPaused (contracts/Token.sol#1006) should be constant  
ProToken.maxBurnFee (contracts/Token.sol#884) should be constant  
ProToken.maxBuybackFee (contracts/Token.sol#886) should be constant  
ProToken.maxExtraSellFee (contracts/Token.sol#889) should be constant  
ProToken.maxLiqFee (contracts/Token.sol#882) should be constant  
ProToken.maxTaxFee (contracts/Token.sol#883) should be constant  
ProToken.maxWalletFee (contracts/Token.sol#885) should be constant  
ProToken.minMxTxPercentage (contracts/Token.sol#887) should be constant  
ProToken.minMxWalletPercentage (contracts/Token.sol#888) should be constant  
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-constant  
  
ProToken._decimals (contracts/Token.sol#949) should be immutable  
ProToken._maxTxAmount (contracts/Token.sol#997) should be immutable  
ProToken._maxWalletAmount (contracts/Token.sol#998) should be immutable  
ProToken._name (contracts/Token.sol#947) should be immutable  
ProToken._symbol (contracts/Token.sol#948) should be immutable  
ProToken._tTotal (contracts/Token.sol#943) should be immutable  
ProToken.burnAutomaticGeneratedLiquidity (contracts/Token.sol#891) should be immutable  
ProToken.buyBackUpperLimit (contracts/Token.sol#1000) should be immutable  
ProToken.canBurn (contracts/Token.sol#1007) should be immutable  
ProToken.canMint (contracts/Token.sol#1004) should be immutable  
ProToken.canPause (contracts/Token.sol#1005) should be immutable  
ProToken.charityFeeToken (contracts/Token.sol#991) should be immutable  
ProToken.feeWalletCharity (contracts/Token.sol#983) should be immutable  
ProToken.hasBlacklist (contracts/Token.sol#1003) should be immutable  
ProToken.pcsV2Pair (contracts/Token.sol#981) should be immutable  
ProToken.pcsV2Router (contracts/Token.sol#980) should be immutable  
ProToken.rewardToken (contracts/Token.sol#940) should be immutable  
ProToken.router (contracts/Token.sol#938) should be immutable  
ProToken.walletCharityFeeInBNB (contracts/Token.sol#987) should be immutable  
ProToken.walletDevFeeInBNB (contracts/Token.sol#988) should be immutable  
ProToken.walletFeeInBNB (contracts/Token.sol#986) should be immutable  
Reference: https://github.com/crytic/slither/wiki/Detector-Documentation#state-variables-that-could-be-declared-immutable
```

**Result => A static analysis of contract's source code has been performed using slither,  
No major issues were found in the output**



# FUNCTIONAL TESTING

---

## 1- Adding liquidity (**passed**):

<https://testnet.bscscan.com/tx/0xbb27b530440f403157b51b2beb6ae098045526ed85acf150de4d6cbf6ef9694>

## 2- Buying when excluded from fees (0% tax) (**passed**):

<https://testnet.bscscan.com/tx/0x5168826cb70dac37254bf19d775c28f68ab32f20a28836f462d4a4b599668425>

## 3- Selling when excluded from fees (0% tax) (**passed**):

<https://testnet.bscscan.com/tx/0xe2d95496e7ba5d4d60a6af9dc6f2a1ca36a2b9245c9347791bca575c1373d404>

## 4- Transferring when excluded from fees (0% tax) (**passed**):

<https://testnet.bscscan.com/tx/0xd35bf2c6cf8a18ec769d06a66569327ec86dfc95fed1ffe2d70fa42fe5fc8035>

## 5- Buying when not excluded from fees (0-60% tax) (**passed**):

<https://testnet.bscscan.com/tx/0x2e93b18a671974d4c923a47ef93f970526735c44cf9f07f86e47e45345fdfb68>

## 6- Selling when not excluded from fees (0-60% tax) (**passed**):

<https://testnet.bscscan.com/tx/0x800125235094de7bf41247bfe4ba64be7a39c5d7db7e2d4ff24b276def345b76>



# FUNCTIONAL TESTING

---

**7- Transferring when not excluded from fees (0-60% tax)**

**(passed):**

<https://testnet.bscscan.com/tx/0x614aef652e1f9e6bce707b89d5f0069cfa8aeeb2c54a7dd0dc9a02b394066175>

**8- Internal swap (passed):**

<https://testnet.bscscan.com/tx/0x800125235094de7bf41247bfe4ba64be7a39c5d7db7e2d4ff24b276def345b76>



# High Risk Issues

## Centralization – Excessive Fees

**Severity:** High

**function:** setAllFeePercent

**Status:** Open

### Overview:

Owner is able to set buy/sell/transfer fees up to 60%

```
function setAllFeePercent(
    uint8 taxFee,
    uint8 liquidityFee,
    uint8 burnFee,
    uint8 marketingFee,
    uint8 devFee,
    uint8 rewardFee,
    uint8 extraSellFee
) external onlyOwner {
    uint8 _maxFee = 10;
    require(taxFee >= 0 && taxFee <= maxTaxFee, "TF err");
    require(liquidityFee >= 0 && liquidityFee <= maxLiqFee, "LF err");
    require(burnFee >= 0 && burnFee <= maxBurnFee, "BF err");
    require(extraSellFee >= 0 && extraSellFee <= maxExtraSellFee, "ESF err");
    require(marketingFee >= 0 && marketingFee <= _maxFee, "WF err");
    require(devFee >= 0 && devFee <= _maxFee, "WFT err");
    require(rewardFee >= 0 && rewardFee <= _maxFee, "RF err");
    //both tax fee and reward fee cannot be set
    require(rewardFee == 0 || taxFee == 0, "RT fee err");
    _walletFee = marketingFee;
    _walletDevFee = devFee;
    _rewardFee = rewardFee;
    _taxFee = taxFee;
    _liquidityFee = liquidityFee;
    _burnFee = burnFee;
    _extraSellFee = extraSellFee;
}
```

### Suggestion

According to pinksale safu criteria, its suggested to have a limitation of max 10% for buy/sell/transfer fees (each one seperately)

0 <= Buy Fees <= 10

0 <= Sell Fees <= 10

0 <= Transfer Fees <= 10



# High Risk Issues

---

## Configuration – Invalid reward tokens

Severity: **High**

**function:** setMarketingFeeToken - setDevFeeToken

**Status:** Open

### Overview:

Owner is able to change marketing and dev reward tokens to any arbitrary address. Setting these addresses to an EOA, or an ERC20 token without liquidity on PCS V2 can disable sells.

```
function setMarketingFeeToken(address feeToken) external onlyOwner {  
    marketingFeeToken = feeToken;  
}
```

```
function setDevFeeToken(address feeToken) external onlyOwner {  
    devFeeToken = feeToken;  
}
```

### Suggestion

Ensure that dev and marketing fee tokens are immutable



# Medium Risk Issue

## Centralization – EOA receiving auto-liquidity generated LP tokens

**Severity:** Medium

**function:** addLiquidity

**Status:** Open

### Overview:

Auto-liquidity generated tokens are sent to an EOA (Externally owned account). This LP tokens can be used to remove a portion of liquidity

```
function addLiquidity(uint256 tokenAmount, uint256 ethAmount) private {
    // approve token transfer to cover all possible scenarios
    _approve(address(this), address(pcsV2Router), tokenAmount);

    address liquidAddr = dead;

    if (!burnAutomaticGeneratedLiquidity) {
        liquidAddr = owner();
    }
    // add the liquidity
    pcsV2Router.addLiquidityETH{value: ethAmount}(
        address(this),
        tokenAmount,
        0, // slippage is unavoidable
        0, // slippage is unavoidable
        liquidAddr,
        block.timestamp
    );
}
```

### Suggestion

It's suggested to Burn or Lock those new LP tokens.



# 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 specialize 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://t.me/Audit_Ace)**



**[https://twitter.com/auditace\\_](https://twitter.com/auditace_)**



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

---