RugFreeCoins Audit

Papa Inu Token
Smart Contract Security Audit

July 21st ,2023

Overview

✅ No mint function found, the owner cannot mint tokens after initial deployment.

✅ The owner can’t set a max transaction limit

✅ The owner can’t pause trading once it’s enabled

❌ The owner must enable trade for the holders, if trading remains disabled, no one would be able to buy and sell.

✅ The owner can't change fees.

✅ The owner can't blacklist wallets.

✅ The owner can’t set a max wallet limit

✅ The owner can’t claim the contract’s balance of its own token.

- **High severity issues**

The owner must enable trade for the holders, if trading remains disabled, no one would be able to buy and sell.

```
function enableTrading() public onlyOwner {
    require(!tradingEnabled, "Trading already enabled!");
    require(hasLiqBeenAdded, "Liquidity must be added.");
    tradingEnabled = true;
    swapEnabled = true;
    allowedPresaleExclusion = false;
}
```

Contents

Audit details

Audited project
Papa Inu Token

Contract Address
0x913cBc1eb0699B86b403E19cAdb482EE5D05DB2b

Client contact
Papa Inu Token Team

Blockchain
Avalanche Smart chain

Project website
https://papainu.io

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 the Papa Inu Token Team to perform an audit of the smart contract.

https://snowtrace.io/address/0x913cbc1eb0699b86b403e19cadb482ee5d05db2b

This audit focuses on verifying 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, and long-term sustainability, and as a guide to improving the smart contract's security posture by remediating the identified issues.

Roadmap

Phase 01

- Website Building & Social Media
- Audit Contract
- Presale On Pinksale
- Launch On Pancake
- Promotional Cooperation With Partners (KOLs, Community, Influencers)
- CoinMarketcap & CoinGecko Listing
- The First Milestone "1,000 Holders"

Phase 02

- Big Partners Announcement
- APAC/Europe/US Marketing
- Papa Inu Ambassador Campaign
- Certik Audit
- Papa NFT Collection Release
- Reach "5,000" Holders

Phase 03

- Papa NFT Limited Edition Released
- Launching Papa Inu Swap
- Top 10 CEX Listing
- (Coinstore, XT, CoinEx, CoinW, MEXC, Bitmart, Kucoin, Bitget, Bybit, Kucoin ...etc)
- Papa Inu Dao - Investment Dao

Tokenomics

5% tax when buying & selling (21/07/2023)

- 5% of trade goes to the marketing wallet in Avax

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 taking part in the Papa Inu token ecosystem.
- Anyone who’s interested in taking part in the future plans of Papa Inu Token.
- Anyone who’s interested in making financial transactions with any other party using Papa Inu Token as the currency.

Potential to grow with score points

1.	Project efficiency	8/10
2.	Project uniqueness	7/10
3	Information quality	8/10
4	Service quality	8/10
5	System quality	8/10
6	Impact on the community	8/10
7	Impact on the business	9/10
8	Preparing for the future	8/10
9	Smart contract security	9/10
10	Smart contract functionality assessment	10/10
Total Points		**8.3/10**

Contract details

Token contract details for 21st of July 2023

Contract name	Papa Inu
Contract address	0x913cBc1eb0699B86b403E19cAdb482EE5D05DB2b
Token supply	100,000,000,000,000,000
Token ticker	PAPA
Decimals	9
Token holders	1
Transaction count	2
Contract deployer address	0x449D540D0A7a256796233c0973278bB1833270Af
Contract ‘s current owner address	0x845ee2cc3bd3961422c02a9423da9de92a9e2067

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 & centralization	Access control of owners	High Issue
		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

The 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 their visibility and mutability.

Contract	**Type**	**Bases**		
L	Function Name	Visibility	Mutability	Modifiers
SafeMath	**Library**			
L	add	Internal 🔒		
L	sub	Internal 🔒		
L	sub	Internal 🔒		
L	mul	Internal 🔒		
L	div	Internal 🔒		
L	div	Internal 🔒		
IERC20	**Interface**			
L	totalSupply	External ❗		NO ❗
L	decimals	External ❗		NO ❗
L	symbol	External ❗		NO ❗

└	name	External ❗		NO ❗
└	getOwner	External ❗		NO ❗
└	balanceOf	External ❗		NO ❗
└	transfer	External ❗	🛑	NO ❗
└	allowance	External ❗		NO ❗
└	approve	External ❗	🛑	NO ❗
└	transferFrom	External ❗	🛑	NO ❗
IJoeSwap Pair	**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 ❗
└	mint	External ❗	🛑	NO ❗

└	burn	External !	🛑	NO !
└	swap	External !	🛑	NO !
└	skim	External !	🛑	NO !
└	sync	External !	🛑	NO !
└	initialize	External !	🛑	NO !
IJoeSwap Router	**Interface**			
└	factory	External !		NO !
└	WAVAX	External !		NO !
└	addLiquidity	External !	🛑	NO !
└	addLiquidityAVAX	External !	💵	NO !
└	removeLiquidity	External !	🛑	NO !
└	removeLiquidityAVAX	External !	🛑	NO !
└	removeLiquidityWithPermit	External !	🛑	NO !
└	removeLiquidityAVAXWithPermit	External !	🛑	NO !
└	swapExactTokensForTokens	External !	🛑	NO !

L	swapTokensForExactTokens	External !		NO !
L	swapExactAVAXForTokens	External !		NO !
L	swapTokensForExactAVAX	External !		NO !
L	swapExactTokensForAVAX	External !		NO !
L	swapAVAXForExactTokens	External !		NO !
L	quote	External !		NO !
L	getAmountOut	External !		NO !
L	getAmountIn	External !		NO !
L	getAmountsOut	External !		NO !
L	getAmountsIn	External !		NO !
L	removeLiquidityAVAXSupportingFeeOnTransferTokens	External !		NO !
L	removeLiquidityAVAXWithPermitSupportingFeeOnTransferTokens	External !		NO !
L	swapExactTokensForTokensSupportingFeeOnTransferTokens	External !		NO !
L	swapExactAVAXForTokensSupportingFeeOnTransferTokens	External !		NO !
L	swapExactTokensForAVAXSupportingFeeOnTransferTokens	External !		NO !

IjoeSwap Factory	**Interface**			
L	feeTo	External !		NO !
L	feeToSetter	External !		NO !
L	migrator	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 !
L	setMigrator	External !	🛑	NO !
Papalnu Token	**Implementation**	**IERC20**		
L		Public !	💵	NO !
L		External !	💵	NO !
L	totalSupply	External !		NO !

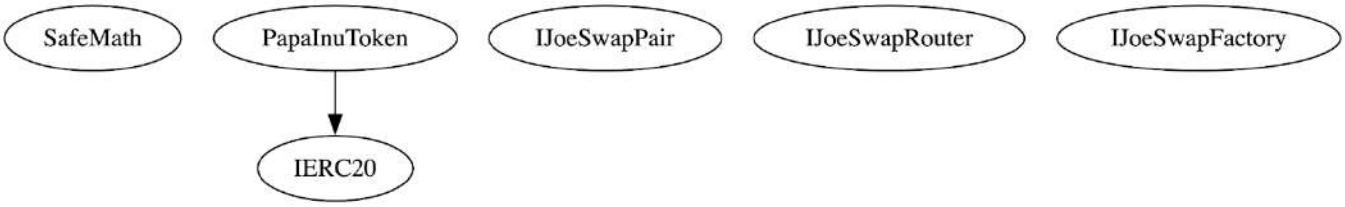
L	decimals	External !		NO !
L	symbol	External !		NO !
L	name	External !		NO !
L	getOwner	External !		NO !
L	allowance	Public !		NO !
L	approve	External !	🛑	NO !
L	_approve	Internal 🔒	🛑	
L	balanceOf	Public !		NO !
L	transfer	Public !	🛑	NO !
L	transferFrom	External !	🛑	NO !
L	_transfer	Internal 🔒	🛑	
L	burn	Public !	🛑	NO !
L	_burn	Internal 🔒	🛑	
L	_hasLimits	Internal 🔒		
L	finalizeTransfer	Internal 🔒	🛑	
L	_transferAmount	Internal 🔒	🛑	

L	_checkLiquidityAdd	Internal 🔒	🛑	
L	manualSwap	External ❗	🛑	NO ❗
L	swapTokensForEth	Private 🔐	🛑	lockThe Swap
L	manualSend	External ❗	🛑	NO ❗
L	sendAVAXToFee	Private 🔐	🛑	
L	approveContractContingency	External ❗	🛑	onlyOwner
L	transferOwner	External ❗	🛑	onlyOwner
L	renounceOwnership	External ❗	🛑	onlyOwner
L	excludePresaleAddresses	External ❗	🛑	onlyOwner
L	setDevAddress	External ❗	🛑	onlyOwner
L	excludeMultipleAccountsFromFees	Public ❗	🛑	onlyOwner
L	setFee	Public ❗	🛑	onlyOwner
L	setSwapNumber	Public ❗	🛑	onlyOwner
L	updateSwapEnabled	External ❗	🛑	onlyOwner
L	enableTrading	Public ❗	🛑	onlyOwner

Legend

Symbol	Meaning
	Function can modify state
	Function is payable

Inheritance Hierarchy



Security issue checking status

- **High severity issues**

The owner must enable trade for the holders, if trading remains disabled, no one would be able to buy and sell.

```
function enableTrading() public onlyOwner {
    require(!tradingEnabled, "Trading already enabled!");
    require(hasLiqBeenAdded, "Liquidity must be added.");
    tradingEnabled = true;
    swapEnabled = true;
    allowedPresaleExclusion = false;
}
```

- **Medium severity issues**
  No medium severity issues found

- **Low severity issues**
  No low- severity issues found

Owner privileges

- Owner and dev wallet can manually swap contract tokens before swap point met

```
function manualSwap() external {
    require(msg.sender == devAddress || msg.sender == _owner);
    uint256 contractBalance = balanceOf(address(this));
    swapTokensForEth(contractBalance);
}
```

- Owner can dev wallet can transfer contract BNB balance to dev wallet

```
function manualSend() external {
    require(msg.sender == devAddress || msg.sender == _owner);
    uint256 contractAVAXBalance = address(this).balance;

    sendAVAXToFee(contractAVAXBalance);
}
```

- Owner can give max approval to contract to spend tokens in contract

```
function approveContractContingency() external onlyOwner returns (bool) {
    _approve(address(this), address(dexRouter), type(uint256).max);
    return true;
}
```

- Owner can enable/disable swapping

```
function updateSwapEnabled(bool _enabled) external onlyOwner {
    swapEnabled = _enabled;
}
```

❖ Owner can transfer ownership

```
function transferOwner(address _newOwner) external onlyOwner {
    require(
        _newOwner != address(0),
        "Call renounceOwnership to transfer owner to the zero address."
    );
    require(
        _newOwner != DEAD,
        "Call renounceOwnership to transfer owner to the zero address."
    );
    if (balanceOf(_owner) > 0) {
        finalizeTransfer(_owner, _newOwner, balanceOf(_owner));
    }

    address oldOwner = _owner;
    _owner = _newOwner;
    _isExcludedFromFee[_owner] = true;

    emit OwnershipTransferred(oldOwner, _newOwner);
}
```

❖ Owner can renounce the ownership

```
    function renounceOwnership() external onlyOwner {
        address oldOwner = _owner;
        _owner = address(0);
        emit OwnershipTransferred(oldOwner, address(0));
    }
```

❖ Owner can enable trading, once enabled can not disable it again

```
function enableTrading() public onlyOwner {
    require(!tradingEnabled, "Trading already enabled!");
    require(hasLiqBeenAdded, "Liquidity must be added.");
    tradingEnabled = true;
    swapEnabled = true;
    allowedPresaleExclusion = false;
}
```

- ❖ Owner can exclude presale address (excluded wallets can do transactions before adding lp and enabling trading)

```
function excludePresaleAddresses(
    address _router,
    address _presale
) external onlyOwner {
    require(allowedPresaleExclusion);
    require(
        _router != address(this) &&
            _presale != address(this) &&
            lpPair != _router &&
            lpPair != _presale,
        "Just don't."
    );
    if (_router == _presale) {
        _liquidityHolders[_presale] = true;
    } else {
        _liquidityHolders[_router] = true;
        _liquidityHolders[_presale] = true;
    }
}
```

- ❖ Owner can change the dev wallet

```
function setDevAddress(address _newAddress) external onlyOwner {
    require(devAddress != address(0), "address cannot be 0");
    devAddress = payable(_newAddress);
}
```

❖ Owner can exclude/include multiple wallets from fees

```
function excludeMultipleAccountsFromFees(
    address[] calldata _accounts,
    bool _excluded
) public onlyOwner {
    require(
        _accounts.length <= 10,
        "The number of accounts cannot be more than 10."
    );
    for (uint256 i = 0; i < _accounts.length; i++) {
        _isExcludedFromFee[_accounts[i]] = _excluded;
    }
}
```

❖ Owner can change buy and sell fees each fee maximum upto 10%

```
function setFee(
    uint256 _taxFeeOnBuy,
    uint256 _taxFeeOnSell
) public onlyOwner {
    require(_taxFeeOnBuy <= 100, "Tax cannot be more than 10.");
    require(_taxFeeOnSell <= 100, "Tax cannot be more than 10.");
    taxFeeOnBuy = _taxFeeOnBuy;
    taxFeeOnSell = _taxFeeOnSell;
}
```

❖ Owner can change the swap poit

```
function setSwapNumber(uint256 _numTokensSellToSwap) public onlyOwner {
    require(
        _numTokensSellToSwap >= 100_000_000 &&
            _numTokensSellToSwap <= 100_000_000_000_000,
        "Swap point is incorrect."
    );
    numTokensSellToSwap = _numTokensSellToSwap * 10 ** _decimals;
}
```

Audit conclusion

RugFreeCoins team has performed in-depth testings, line-by-line manual code review, and automated audit of the smart contract. The smart contract was analyzed mainly for common smart contract vulnerabilities, exploits, manipulations, and hacks. According to the smart contract audit.

Smart contract functional Status: **PASS**

Number of risk issues: **1**

Solidity code functional issue level: **PASS**

Number of owner privileges: **12**

Centralization risk correlated to the active owner: **HIGH**

Smart contract active ownership: **ACTIVE**