



RugFreeCoins Audit



Pepe Le Pew Coin Smart Contract Security Audit

June 29th ,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. **Resolved: Trade enables function is already called & ownership renounced.**
- ✔ 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.

Contents

Overview	ii
Audit details	1
Disclaimer	2
Background	3
Potential to grow with score points	5
Total Points	5
Target market and the concept	6
Contract details	7
Contract code function details	8
Contract description table	10
Security issue checking status	13
Owner privileges	14
Audit conclusion	16

Audit details



Audited project
Pepe Le Pew Coin



Contract Address
0xa34Ee6108Fe427f91edce0D6520d9fEc0E64F67b



Client contact
Pepe Le Pew CoinTeam



Blockchain
Ethereum



Project website
www.pepelepewcoin.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 the Pepe Le Pew Coin Team to perform an audit of the smart contract.

<https://etherscan.io/token/0xa34Ee6108Fe427f91edce0D6520d9fEc0E64F67b>

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.

Tokenomics

0% tax when buying and selling

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
9	Smart contract security	10/10
10	Smart contract functionality assessment	10/10
Total Points		10/10

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 Pepe Le Pew Coin ecosystem.
- Anyone who's interested in taking part in the future plans of Pepe Le Pew Coin.
- Anyone who's interested in making financial transactions with any other party using Pepe Le Pew Coin as the currency.

Contract details

Token contract details for 29th of June 2023

Contract name	Pepe Le Pew Coin
Contract address	0xa34Ee6108Fe427f91edce0D6520d9fEc0E64F67b
Token supply	165,165,125,165,230
Token ticker	\$PLPC
Decimals	9
Token holders	395
Transaction count	662
Contract deployer address	0xe5ea5e472B76C6542847501d48C02B4D451E1A57
Contract's current owner address	0xe5ea5e472B76C6542847501d48C02B4D451E1A57











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	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

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
PLPC	Implementation	IERC20, Ownable		
L		Public !		NO !
L		External !		NO !
L	totalSupply	External !		NO !
L	name	Public !		NO !
L	symbol	Public !		NO !
L	decimals	Public !		NO !
L	balanceOf	Public !		NO !
L	allowance	External !		NO !
L	approve	Public !		NO !
L	_approve	Internal 		
L	approveMax	External !		NO !
L	transfer	External !		NO !
L	transferFrom	External !		NO !
L	_transferFrom	Internal 		

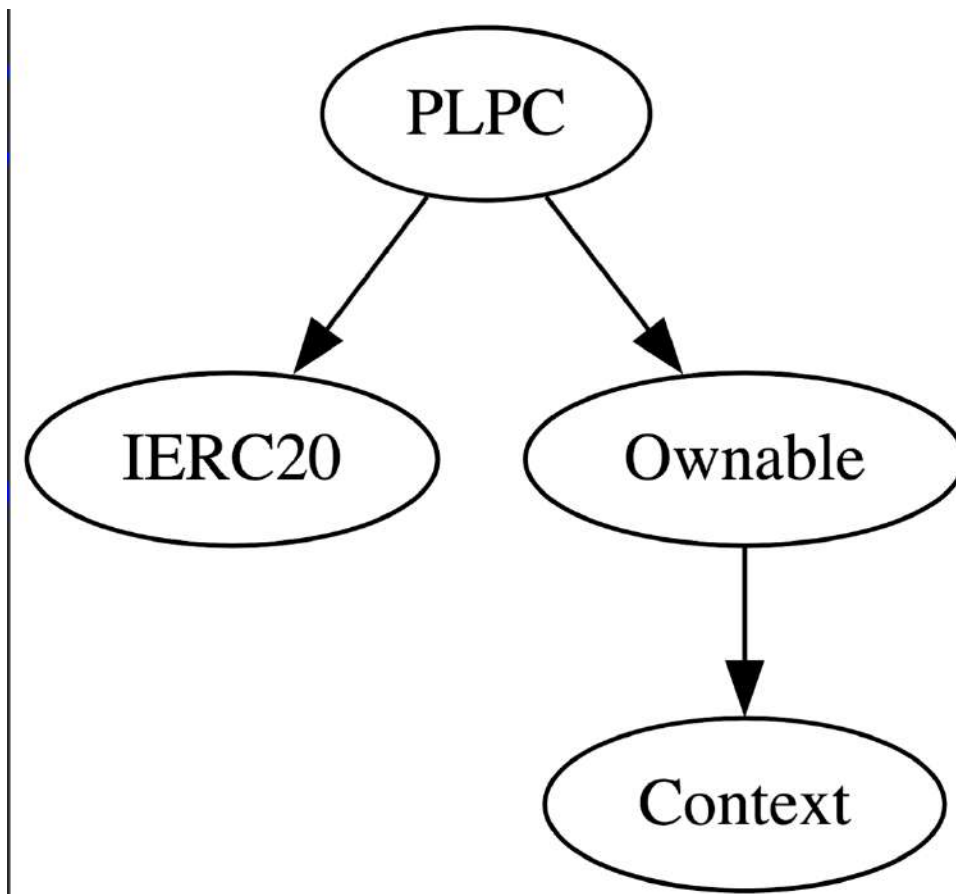
L	enableTrading	External !		onlyOwner
L	setAuthorizedWallets	External !		onlyOwner
L	rescueETH	External !		onlyOwner
L	burnTokens	Public !		NO !
L	changeBurnSettings	External !		NO !
L	changeBurnManager	External !		onlyOwner
Ownable	Implementation	Context		
L		Public !		NO !
L	owner	Public !		NO !
L	_checkOwner	Internal 		
L	renounceOwnership	Public !		onlyOwner
L	transferOwnership	Public !		onlyOwner
L	_transferOwnership	Internal 		
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 !
Context	Implementation			
L	_msgSender	Internal 🔒		
L	_msgData	Internal 🔒		

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

❖ **Centralization Risk**

No Centralization issues found

Owner privileges

- ❖ Owner can enable trading once enabled can not disable again

```
function enableTrading() external onlyOwner {  
    require(!isTradeEnabled, "Trading already enabled");  
    isTradeEnabled = true;  
    nextBurnTime = block.timestamp + burnIntevel;  
}
```

- ❖ Owner can authorize wallets (who can do transactions before enabling trade)

```
function setAuthorizedWallets(  
    address _wallet,  
    bool _status  
) external onlyOwner {  
    isAuthorized[_wallet] = _status;  
}
```

- ❖ Owner can get ETH from the token contract

```
function rescueETH() external onlyOwner {  
    uint256 balance = address(this).balance;  
    require(balance > 0, "No enough ETH to transfer");  
  
    payable(msg.sender).transfer(balance);  
}
```

- ❖ Burn Manager can change burn frequency and burning percentage

```
function changeBurnSettings(  
    uint256 _burnIntevel,  
    uint256 _percentage  
) external {  
    require(  
        msg.sender == burnManager,  
        "Only Burn Manager can call this function"  
    );  
  
    burnIntevel = _burnIntevel;  
    burnPercentage = _percentage;  
}
```

- ❖ Owner can change burn manager

```
function changeBurnManager(address _burnManager) external onlyOwner {  
    burnManager = _burnManager;  
}
```

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: **0**

Solidity code functional issue level: **PASS**

Number of owner privileges: **5**

Centralization risk correlated to the active owner: **NONE**

Smart contract active ownership: **RENOUNCED**