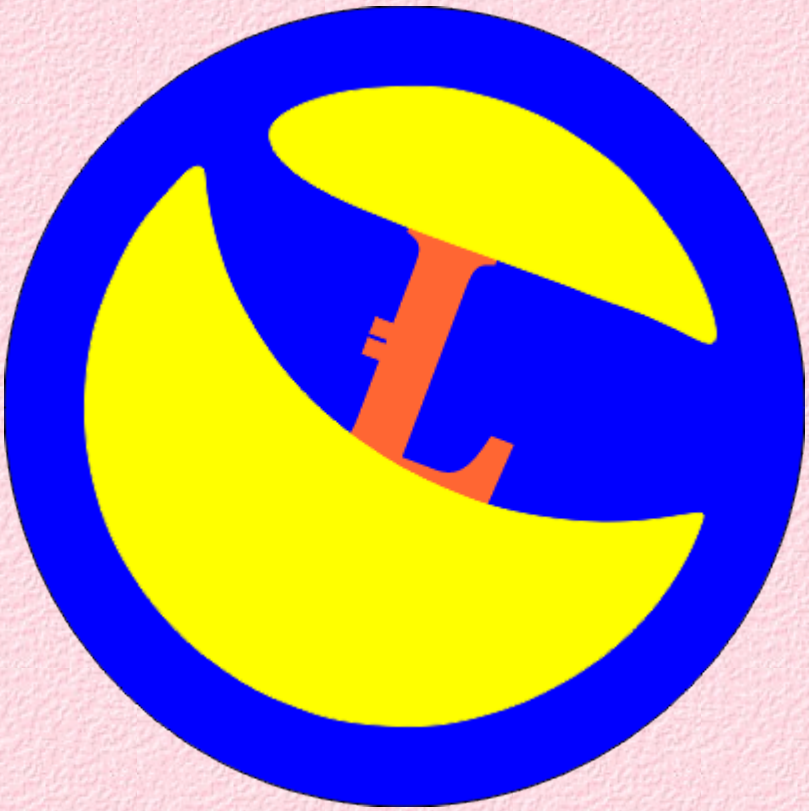
PinkAudit.AI

Date: 06.03.2023

Smart Contract Security Audit

Lunamunt



Telegram: contact: PinkAudit.AI

Table of Contents

Audit Result

Findings

ERCs: ERC20, ERC2612

Vulnerability Check ERC20

Vulnerability Check events

Vulnerability Check DividendPayingToken

Vulnerability Check _LUNADividendTracker

Vulnerability Check Lunamunt

Overview

Metrics

Contract Details

Purpose

Project Summary

Conclusion

Disclaimer

PinkAuditAI

Audit Result

√ Lunamunt has passed the smart contract assessment with below listed privileges.

Audit Result	PASSED √
Ownership	Not Renounced yet √
KYC Verification	Not at date of report √
Audit Date	06-03-2023 √
Audit Team	PinkAudit √

Findings

During the audit, the following issues were identified: (Important issues listed below).

- 46 low issues
- 13 medium issues
- 3 high issues

Auto liquidity is going to an externally owned account	⚠
Owner can exclude accounts from rewards	⚠
Owner can exclude an account from paying fees	⚠
Owner can change the fees but with limit of 24% at max	⚠
Trading must be enabled by the owner	⚠
Owner can change max transaction amount within reasonable limits	⚠
Owner can change max wallet token amount within reasonable limits	⚠
Owner can change swap settings	⚠
Owner can withdraw any token from the contract	⚠
Repeated function available	⚠

After a manual check, we found that all the issues above specifically the (high issues) were not critical, however as an investor you must take notice of the above mentioned actions:

PinkAuditAI

ERCs: ERC20, ERC2612

Number of lines: 1469 (+ 0 in dependencies, + 0 in tests)
Number of assembly lines: 0
Number of contracts: 17 (+ 0 in dependencies, + 0 tests)

Number of low issues: 46
Number of medium issues: 13
Number of high issues: 3

Name	# Functions	ERCS	ERC20 Info	Complex Code	Features
IUniswapV2Factory	8			No	
IUniswapV2Pair	27	ERC20,ERC2612	∞ Minting Approve Race Cond.	No	
IUniswapV2Router02	14			No	Receive ETH
IterableMapping	6			No	
SafeMath	13			No	
SafeMathInt	5			No	
SafeMathUint	1			No	
_LUNADividendTracker	66	ERC20	No Minting Approve Race Cond.	Yes	Tokens interaction
Lunamunt	89	ERC20	No Minting Approve Race Cond.	Yes	Receive ETH Send ETH Tokens interaction

Vulnerability Check ERC20

Check functions
[√] totalSupply() is present
    [√] totalSupply() -> (uint256) (correct return type)
    [√] totalSupply() is view
[√] balanceOf(address) is present
    [√] balanceOf(address) -> (uint256) (correct return type)
    [√] balanceOf(address) is view
[√] transfer(address,uint256) is present
    [√] transfer(address,uint256) -> (bool) (correct return type)
    [√] Transfer(address,address,uint256) is emitted
[√] transferFrom(address,address,uint256) is present
    [√] transferFrom(address,address,uint256) -> (bool) (correct return type)
    [√] Transfer(address,address,uint256) is emitted
[√] approve(address,uint256) is present
    [√] approve(address,uint256) -> (bool) (correct return type)
    [√] Approval(address,address,uint256) is emitted
[√] allowance(address,address) is present
    [√] allowance(address,address) -> (uint256) (correct return type)
    [√] allowance(address,address) is view
[√] name() is present
    [√] name() -> (string) (correct return type)
    [√] name() is view
[√] symbol() is present
    [√] symbol() -> (string) (correct return type)
    [√] symbol() is view
[√] decimals() is present
    [√] decimals() -> (uint8) (correct return type)
    [√] decimals() is view

Vulnerability Check events

[√] Transfer(address,address,uint256) is present
    [√] parameter 0 is indexed
    [√] parameter 1 is indexed
[√] Approval(address,address,uint256) is present
    [√] parameter 0 is indexed
    [√] parameter 1 is indexed

[√] The check passed successfully without any vulnerabilities.

PinkAuditAI

Vulnerability Check DividendPayingToken

Check functions
[√] totalSupply() is present
    [√] totalSupply() -> (uint256) (correct return type)
    [√] totalSupply() is view
[√] balanceOf(address) is present
    [√] balanceOf(address) -> (uint256) (correct return type)
    [√] balanceOf(address) is view
[√] transfer(address,uint256) is present
    [√] transfer(address,uint256) -> (bool) (correct return type)
    [ ] Must emit be view Transfer(address,address,uint256)
[√] transferFrom(address,address,uint256) is present
    [√] transferFrom(address,address,uint256) -> (bool) (correct return type)
    [ ] Must emit be view Transfer(address,address,uint256)
[√] approve(address,uint256) is present
    [√] approve(address,uint256) -> (bool) (correct return type)
    [√] Approval(address,address,uint256) is emitted
[√] allowance(address,address) is present
    [√] allowance(address,address) -> (uint256) (correct return type)
    [√] allowance(address,address) is view
[√] name() is present
    [√] name() -> (string) (correct return type)
    [√] name() is view
[√] symbol() is present
    [√] symbol() -> (string) (correct return type)
    [√] symbol() is view
[√] decimals() is present
    [√] decimals() -> (uint8) (correct return type)
    [√] decimals() is view

Check events
[√] Transfer(address,address,uint256) is present
    [√] parameter 0 is indexed
    [√] parameter 1 is indexed
[√] Approval(address,address,uint256) is present
    [√] parameter 0 is indexed
    [√] parameter 1 is indexed

[√] The check passed successfully without any vulnerabilities.

PinkAuditAI

Vulnerability Check _LUNADividendTracker

Check functions
[√] totalSupply() is present
	[√] totalSupply() -> (uint256) (correct return type)
	[√] totalSupply() is view
[√] balanceOf(address) is present
	[√] balanceOf(address) -> (uint256) (correct return type)
	[√] balanceOf(address) is view
[√] transfer(address,uint256) is present
	[√] transfer(address,uint256) -> (bool) (correct return type)
	[ ] Must emit be view Transfer(address,address,uint256)
[√] transferFrom(address,address,uint256) is present
	[√] transferFrom(address,address,uint256) -> (bool) (correct return type)
	[ ] Must emit be view Transfer(address,address,uint256)
[√] approve(address,uint256) is present
	[√] approve(address,uint256) -> (bool) (correct return type)
	[√] Approval(address,address,uint256) is emitted
[√] allowance(address,address) is present
	[√] allowance(address,address) -> (uint256) (correct return type)
	[√] allowance(address,address) is view
[√] name() is present
	[√] name() -> (string) (correct return type)
	[√] name() is view
[√] symbol() is present
	[√] symbol() -> (string) (correct return type)
	[√] symbol() is view
[√] decimals() is present
	[√] decimals() -> (uint8) (correct return type)
	[√] decimals() is view

Check events
[√] Transfer(address,address,uint256) is present
	[√] parameter 0 is indexed
	[√] parameter 1 is indexed
[√] Approval(address,address,uint256) is present
	[√] parameter 0 is indexed
	[√] parameter 1 is indexed

Check _LUNADividendTracker

Check functions
[√] totalSupply() is present
	[√] totalSupply() -> (uint256) (correct return type)
	[√] totalSupply() is view
[√] balanceOf(address) is present
	[√] balanceOf(address) -> (uint256) (correct return type)
	[√] balanceOf(address) is view
[√] transfer(address,uint256) is present
	[√] transfer(address,uint256) -> (bool) (correct return type)
	[ ] Must emit be view Transfer(address,address,uint256)

[√] transferFrom(address,address,uint256) is present
    [√] transferFrom(address,address,uint256) -> (bool) (correct return type)
    [ ] Must emit be view Transfer(address,address,uint256)
[√] approve(address,uint256) is present
    [√] approve(address,uint256) -> (bool) (correct return type)
    [√] Approval(address,address,uint256) is emitted
[√] allowance(address,address) is present
    [√] allowance(address,address) -> (uint256) (correct return type)
    [√] allowance(address,address) is view
[√] name() is present
    [√] name() -> (string) (correct return type)
    [√] name() is view
[√] symbol() is present
    [√] symbol() -> (string) (correct return type)
    [√] symbol() is view
[√] decimals() is present
    [√] decimals() -> (uint8) (correct return type)
    [√] decimals() is view

Check events
[√] Transfer(address,address,uint256) is present
    [√] parameter 0 is indexed
    [√] parameter 1 is indexed
[√] Approval(address,address,uint256) is present
    [√] parameter 0 is indexed
    [√] parameter 1 is indexed

[√] The check passed successfully without any vulnerabilities.

PinkAuditAI

Vulnerability Check Lunamunt

Check functions
[√] totalSupply() is present
    [√] totalSupply() -> (uint256) (correct return type)
    [√] totalSupply() is view
[√] balanceOf(address) is present
    [√] balanceOf(address) -> (uint256) (correct return type)
    [√] balanceOf(address) is view
[√] transfer(address,uint256) is present
    [√] transfer(address,uint256) -> (bool) (correct return type)
    [√] Transfer(address,address,uint256) is emitted
[√] transferFrom(address,address,uint256) is present
    [√] transferFrom(address,address,uint256) -> (bool) (correct return type)
    [√] Transfer(address,address,uint256) is emitted
[√] approve(address,uint256) is present
    [√] approve(address,uint256) -> (bool) (correct return type)
    [√] Approval(address,address,uint256) is emitted
[√] allowance(address,address) is present
    [√] allowance(address,address) -> (uint256) (correct return type)
    [√] allowance(address,address) is view
[√] name() is present
    [√] name() -> (string) (correct return type)
    [√] name() is view
[√] symbol() is present
    [√] symbol() -> (string) (correct return type)
    [√] symbol() is view
[√] decimals() is present
    [√] decimals() -> (uint8) (correct return type)
    [√] decimals() is view

Check events
[√] Transfer(address,address,uint256) is present
    [√] parameter 0 is indexed
    [√] parameter 1 is indexed
[√] Approval(address,address,uint256) is present
    [√] parameter 0 is indexed
    [√] parameter 1 is indexed

    [√] ERC20 has increaseAllowance(address,uint256)
    [√] DividendPayingToken has increaseAllowance(address,uint256)
    [√] _LUNADividendTracker has increaseAllowance(address,uint256)
    [√] _LUNADividendTracker has increaseAllowance(address,uint256)
    [√] Lunamunt has increaseAllowance(address,uint256)

PinkAuditAI

Overview

This audit report was conducted on a set of 17 smart contracts that were compiled with solc. The contracts include ERC20 and ERC2612 tokens, as well as other contracts related to Uniswap V2 and a dividend tracker.

Metrics

The following metrics were gathered during the audit:

- Number of lines: 1469 (+ 0 in dependencies, + 0 in tests)
- Number of assembly lines: 0
- Number of contracts: 17 (+ 0 in dependencies, + 0 tests)
- Number of low issues: 46
- Number of medium issues: 13
- Number of high issues: 3
- ERCs: ERC20, ERC2612

Contract Details

The contracts audited include the following:

- IUniswapV2Factory: 8 functions, no ERCs, no complex code, no special features
- IUniswapV2Pair: 27 functions, ERC20 and ERC2612 tokens, infinite minting, approve race condition, no complex code, no special features
- IUniswapV2Router02: 14 functions, no ERCs, no complex code, receive ETH feature
- IterableMapping: 6 functions, no ERCs, no complex code, no special features
- SafeMath: 13 functions, no ERCs, no complex code, no special features
- SafeMathInt: 5 functions, no ERCs, no complex code, no special features
- SafeMathUint: 1 function, no ERCs, no complex code, no special features
- _LUNADividendTracker: 66 functions, ERC20 token, no minting, approve race condition, complex code, tokens interaction feature
- Lunamunt: 89 functions, ERC20 token, no minting, approve race condition, complex code, receive/send ETH and tokens interaction features

Purpose

The purpose of the audit was to achieve the following:

● Ensure that the smart contract functions as intended.
● Identify potential security issues with the smart contract.
The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by re mediating the issues that were identified.

PinkAuditAI

Project Summary

Token Name LUNAMUNT

Web Site http://lunamunt.com/

Twitter https://twitter.com/Lunamunt

Platform Binance Smart Chain

Token Type BEP20

Language Solidity

Platforms & Tools Remix IDE, Truffle, Ganache, Solhint, VScode, Slither

Contract Address 0x4624BAa18889cf1ecdB777Bd456BF6a8Ab2F7051

Contract Link https://bscscan.com/address/0x4624BAa18889cf1ecdB777Bd456BF6a8Ab2F7051

Testnet Link https://testnet.bscscan.com/address/0x48bc7e00f750903d16ca5ccd750599cccfbe9038

The files:
Lunamunt.sol

Conclusion

Overall, the smart contracts audited were well-designed and implemented. While some optimization and code quality issues were identified, these were not critical, however we do not provide guarantee of future outcomes. We have used all the latest static tools and manual observations to cover maximum possible test cases to scan everything.

Security state of the reviewed contract is “ Well Secured”.
✔ No volatile code.
✔ No high severity issues were found.
✔ Low (or very low) level issues have been fixed.

PinkAuditAI

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as of 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 the team on the basis of what it says or doesn't say, or how team produced it, and it is important for you to conduct your own independent investigations before making any decisions. team go into more detail on this in the below disclaimer below – please make sure to read it in full.

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 PinkAudit.AI and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (PinkAudit.AI's) owe no duty of care towards you or any other person, nor does PinkAudit.AI 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 PinkAudit.AI 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, PinkAudit.AI hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against PinkAudit.AI, 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.