INSPECTOR LOVELY

SMART CONTRACT
SECURITY AUDIT

METAPLE

Scan and check this report
was created at INSPECTOR LOVELY

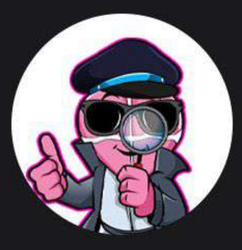
July, 2022

https://inspector.lovely.finance

INSPECTOR LOVELY

Table of Contents

Table of Contents	2
Disclaimer	3
Procedure	4
Terminology	5
Limitations	5
Token Contract Details for 04.07.2022	6
Audit Details	7
Contract Analytics	7
MLX Token Distribution	8
Swap Analysis	8
Contract Analysis	8
Holder Anlaysis	9
Contract Function Details	11
Vulnerabilities checking	12
Security Issues	13
Conclusion	14



INSPECTOR LOVELY

Disclaimer

This is a comprehensive report based on our automated and manual examination of cybersecurity vulnerabilities and framework flaws. We took into consideration smart contract based algorithms, as well. Reading the full analysis report is essential to build your understanding of project's security level. It is crucial to take note, though we have done our best to perform this analysis and report, that you should not rely on the our research and cannot claim what it states or how we created it. Before making any judgments, you have to conduct your own independent research. We will discuss this in more depth in the following disclaimer - please read it fully.

DISCLAIMER: You agree to the terms of this disclaimer by reading this report or any portion thereof. Please stop reading this report and remove and delete any copies of this report that you download and/or print if you do not agree to these conditions. This report is for non-reliability information only and does not represent investment advice. No one shall be entitled to depend on the report or its contents, and INSPECTOR LOVELY and its affiliates shall not be held responsible to you or anyone else, nor shall INSPECTOR LOVELY provide any guarantee or representation to any person with regard to the accuracy or integrity of the report. Without any terms, warranties or other conditions other than as set forth in that exclusion and INSPECTOR LOVELY excludes hereby all representations, warrants, conditions and other terms (including, without limitation, guarantees implied by the law of satisfactory quality, fitness for purposes and the use of reasonable care and skills). The report is provided as "as is" and does not contain any terms and conditions. Except as legally banned. INSPECTOR LOVELY disclaims all responsibility and responsibilities and no claim against INSPECTOR LOVELY is made to any amount or type of loss or damages (without limitation, direct, indirect, special, punitive, consequential or pure economic loses or losses) that may be caused by you or any other person, or any damages or damages, including without limitations (whether innocent or negligent).

Security analysis is based only on the smart contracts. No applications or operations were reviewed for security. No product code has been reviewed

Procedure

Our analysis contains following steps:

1. Project Analysis;

2. Manual analysis of smart contracts:

- Deploying smart contracts on any of the network(Ropsten/Rinkeby) using Remix IDE
- Hashes of all transaction will be recorded
- Behaviour of functions and gas consumption is noted, as well.

3. Unit Testing:

- Smart contract functions will be unit tested on multiple parameters and under multiple conditions to ensure that all paths of functions are functioning as intended.
- In this phase intended behaviour of smart contract is verified.
- In this phase, we would also ensure that smart contract functions are not consuming unnecessary gas.
- Gas limits of functions will be verified in this stage.

4. Automated Testing:

- Mythril
- Oyente
- Manticore
- Solgraph

Terminology

We categorize the finding into 4 categories based on their vulnerability:

- Low-severity issue - less important, must be analyzed
- Medium-severity issue - important, needs to be analyzed and fixed
- High-severity issue - important, might cause vulnerabilities, must be analyzed and fixed
- Critical-severity issue - serious bug causes, must be analyzed and fixed.

Limitations

The security audit of Smart Contract cannot cover all vulnerabilities. Even if no vulnerabilities are detected in the audit, there is no guarantee that future smart contracts are safe. Smart contracts are in most cases safeguarded against specific sorts of attacks. In order to find as many flaws as possible we carried out a comprehensive smart contract audit. Audit is a document that is not legally binding and guarantees nothing.

INSPECTOR LOVELY

Token Contract Details for 04.07.2022

Contract Name: **METAPLE**

Deployed address: **0xeBDA6aaE3f8c96eAEbe33cEEbbAE24Fc39F315d6**

Total Supply: **150,000,000**

Token Tracker: **MLX**

Decimals: **18**

Token holders: **1**

Transactions count: **1**

Top 100 holders dominance: **100.00%**

Audit Details

Project Name: **METAPLE**

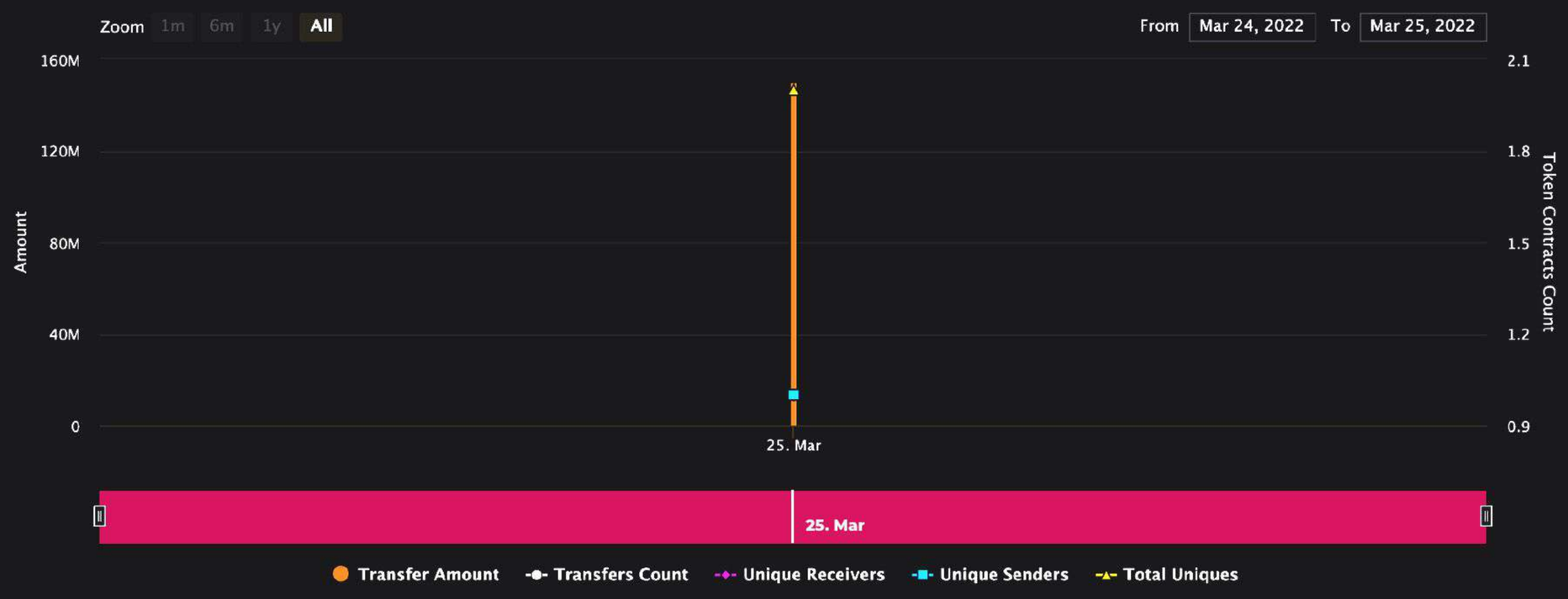
Language: **Solidity**

Compiler Version: **v0.8.11**

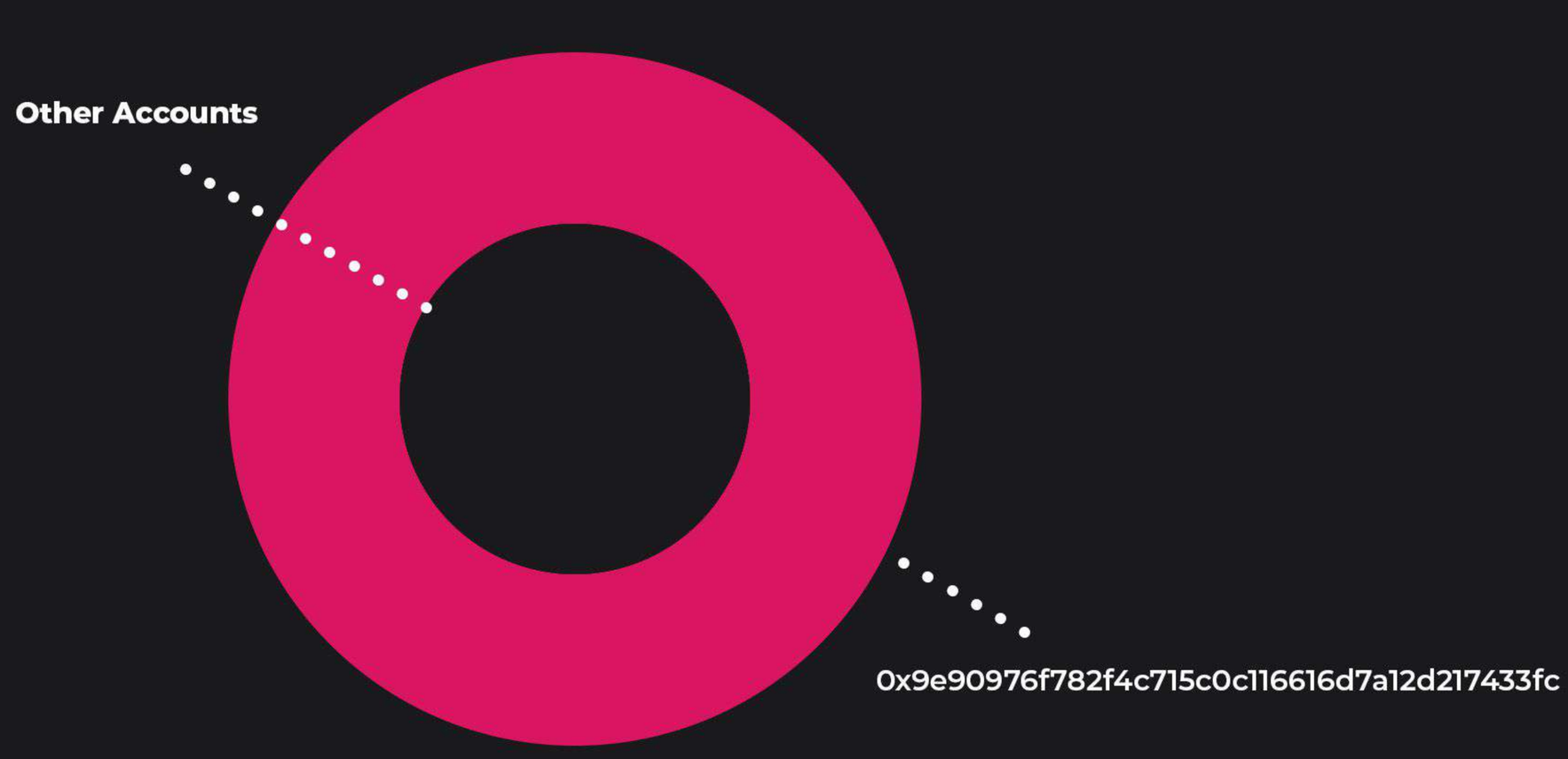
Blockchain: **BSC**

INSPECTOR LOVELY

Contract Analytics



MLX Token Distribution



MLX Top Holders

Rank	Address	Quantity (Token)	Percentage
1	0x9e90976f782f4c715c0c116616d7a12d217433fc	150,000,000	100.0000%

INSPECTOR LOVELY

Swap Analysis

✅ Token is sellable (not a honeypot) at this time

✅ Buy fee is <= 10% (0%)

✅ Sell fee is <= 10% (0%)

Contract Analysis

✅ Verified contract source

❌ Ownership renounced or source does not contain an owner contract.

Holder Analysis

✅ Owner wallet contains less than 5% of circulating token supply (0%)

INSPECTOR LOVELY

Contract Function Details

+ mlx.sol
- Ext initialMint
- [Ext] initReferral

+ BEP20.sol
- [Int] initialMint
- [Ext] getOwner
- [Ext] decimals
- [Ext] symbol
- [Int] getSymbol
- Ext name
- [Ext] totalSupply
- [Ext] balanceOf
- [Ext] transfer
- [Ext] allowance
- [Ext] approve
- [Ext] transferFrom
- [Ext] increaseAllowance
- [Ext] decreaseAllowance
- [Ext] mint
- [Ext] mintMLX
- [Int] transfer
- llnt_mint
- [Intl burn
- [Int] _approve
- [Ext] _burnFrom

+ Utils360.sol
- [Int] append
- [Int] appendADDR
- [Int] appendINT

+ Ref360
- [Ext] referrer
- [Ext] getReferrals
- [Int] _setReferrer

+ IBEP20.sol
- [Ext] totalSupply
- [Ext] decimals
- [Ext] symbol
- [Ext] name
- [Ext] getOwner
- [Ext] balanceOf
- [Ext] transfer
- [Ext] allowance
- [Ext] approve
- [Ext] transferFrom

+ Context.sol
- [Int] _msgSender
- [Int] _msgData

INSPECTOR LOVELY

+ Ownable.sol
- [Pub owner
- [Ext] renounceOwnership
- [Ext] transferOwnership
- [Intl transferOwnership

+ SafeMath.sol
- [Intl add
- Intl sub
- [Intl sub
- [Int] mul
- [Int] div
- [Int] mod
- [Int] mod

INSPECTOR LOVELY

Vulnerabilities checking

Issue Description	Checking Status
Compiler Errors	Completed
Delays in Data Delivery	Completed
Re-entrancy	Completed
Transaction-Ordering Dependence	Completed
Timestamp Dependence	Completed
Shadowing State Variables	Completed
DoS with Failed Call	Completed
DoS with Block Gas Limit	Completed
Outdated Complier Version	Completed
Assert Violation	Completed
Use of Deprecated Solidity Functions	Completed
Integer Overflow and Underflow	Completed
Function Default Visibility	Completed
Malicious Event Log	Completed
Math Accuracy	Completed
Design Logic	Completed
Fallback Function Security	Completed
Cross-function Race Conditions	Completed
Safe Zeppelin Module	Completed

INSPECTOR LOVELY

Owner Privileges

The contract contains ownership functionality and ownership is not renounced which allows the creator or current owner to modify contract behaviour (for example, disable selling or mint new tokens).

Conclusion

Low-severitv issues exist within smart contracts. Smart contracts are free from any critical or high-severity issues.

NOTE: Please check the disclaimer above and note, that audit makes no statements or warranties on business model, investment attractiveness or code sustainability.

INSPECTOR LOVELY

INFO

Website: inspector.lovely.finance
Telegram Community: https://t.me/inspectorlovely

Contact us:
Telegram: t.me/lovelyecosystem
Mail: Info@lovely.finance

INSPECTOR LOVELY

https://inspector.lovely.finance