VITAL BLOCK

SMART CONTRACT AUDIT

@Vital-Block

@VB_Audit

info@vitalblock.org

www.vitalblock.org

PREPARED FOR:

LEONICORN SWAP

INTRODUCTION

Auditing Firm	**VITAL BLOCK SECURITY**
Client Firm	**LEONICORN SWAP**
Methodology	**Automated Analysis, Manual Code Review**
Language	**.Move**
Contract	**0xc8f5f290fe0d155933cfc2cdd13ae11ba97abb4f413770091c3ab70cacc6adae**
Blockchain	**Aptos Blockchain**
Centralization	**Active ownership**
Website	**https://dex.leonicornswap.com**
Discord	**https://discord.gg/bG9RqyGKwE**
Twitter	**https://twitter.com/swapleonicorn**
GitHub	**https://github.com/Leonicornswap**
Prelim Report Date	**December 05, 2022**
Final Report Date	**December 06, 2022**

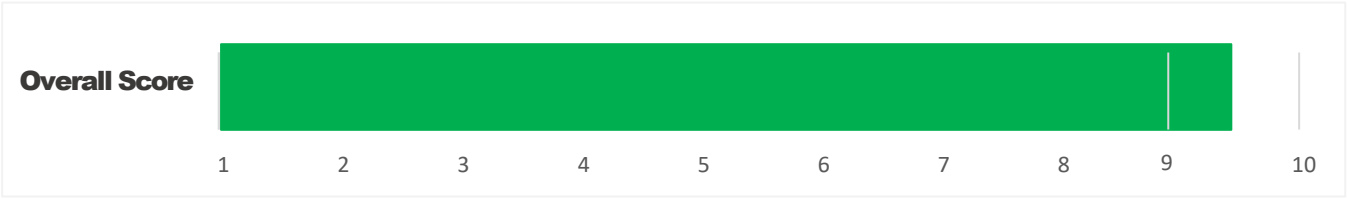
Verify the authenticity of this report on our GitHub Repo: https://www.github.com/vital-block

EXECUTIVE SUMMARY

Vital Block has performed the automated and manual analysis of the Move code. The code was reviewed for common contract vulnerabilities and centralized exploits. Here's a quick audit summary:

Status	Critical ! 🔴	Major " 🟠	Medium # 🟡	Minor $ 🟢	Unknown % 🟤
Open	0	0	0	2	0
Acknowledged	0	0	0	2	0
Resolved	0	0	0	0	0
Noteworty only owner Privileges	Set Taxes and Ratios, Airdrop, Set Protection Settings, Set Reward Properties, Set Reflector Settings, Set Swap Settings, Set Pair and Router				

LEONICORN SWAP smart contract has achieved the following score: 9.5.



ℹ **Please note that smart contracts deployed on blockchains aren't resistant to exploits, vulnerabilities and/or hacks. Blockchain and cryptography assets utilize new and emerging technologies. These technologies present a high level of ongoing risks. For a detailed understanding of risk severity, source code vulnerability, and audit limitations, kindly review the audit report thoroughly.**

ℹ **Please note that centralization privileges regardless of their inherited risk status - constitute an elevated impact on smart contract safety and security.**

TABLE OF CONTENTS

SCOPE OF WORK

Vital Block was consulted by LEONICORN SWAP to conduct the smart contract audit of its MOVE source code. The audit scope of work is strictly limited to mentioned MOVE file only:

- **AbelCoin.move**

External contracts and/or interfaces dependencies are not checked due to being out of scope.

Verify audited contract's contract address and deployed link below:

Public Contract Link	
0xc8f5f290fe0d155933cfc2cdd13ae11ba97abb4f413770091c3ab70cacc6adae	
CONTRACT NAME	AGGREGATOR
LOCKING	TOKENS
DEX TYPE	STAKING, FARMS, POOLS

AUDIT METHODOLOGY

Smart contract audits are conducted using a set of standards and procedures. Mutual collaboration is essential to performing an effective smart contract audit. Here's a brief overview of Vital Block auditing process and methodology:

CONNECT

- **The onboarding team gathers source codes, and specifications to make sure we understand the size, and scope of the smart contract audit.**

AUDIT

- **Automated analysis is performed to identify common contract vulnerabilities. We may use the following third-party frameworks and dependencies to perform the automated analysis:**
  - **Remix IDE Developer Tool**
  - **Open Zeppelin Code Analyzer**
  - **SWC Vulnerabilities Registry**
  - **DEX Dependencies, e.g., Pancakeswap, Uniswap**
- **Simulations are performed to identify centralized exploits causing contract and/or trade locks.**
- **A manual line-by-line analysis is performed to identify contract issues and centralized privileges. We may inspect below mentioned common contract vulnerabilities, and centralized exploits:**

Centralized Exploits	○ **Token Supply Manipulation** ○ **Access Control and Authorization** ○ **Assets Manipulation** ○ **Ownership Control** ○ **Liquidity Access** ○ **Stop and Pause Trading** ○ **Ownable Library Verification**

Common Contract Vulnerabilities	○ Integer Overflow ○ Lack of Arbitrary limits ○ Incorrect Inheritance Order ○ Typographical Errors ○ Requirement Violation ○ Gas Optimization ○ Coding Style Violations ○ Re-entrancy ○ Third-Party Dependencies ○ Potential Sandwich Attacks ○ Irrelevant Codes ○ Divide before multiply ○ Conformance to Solidity Naming Guides ○ Compiler Specific Warnings ○ Language Specific Warnings

REPORT

- The auditing team provides a preliminary report specifying all the checks which have been performed and the findings thereof.
- The client's development team reviews the report and makes amendments to the codes.
- The auditing team provides the final comprehensive report with open and unresolved issues.

PUBLISH

- The client may use the audit report internally or disclose it publicly.

ℹ It is important to note that there is no pass or fail in the audit, it is recommended to view the audit as an unbiased assessment of the safety of solidity codes.

RISK CATEGORIES

Smart contracts are generally designed to hold, approve, and transfer tokens. This makes them very tempting attack targets. A successful external attack may allow the external attacker to directly exploit. A successful centralization-related exploit may allow the privileged role to directly exploit. All risks which are identified in the audit report are categorized here for the reader to review:

Risk Type	Definition
Critical !	These risks could be exploited easily and can lead to asset loss, data loss, asset, or data manipulation. They should be fixed right away.
Major "	These risks are hard to exploit but very important to fix, they carry an elevated risk of smart contract manipulation, which can lead to high-risk severity.
Medium #	These risks should be fixed, as they carry an inherent risk of future exploits, and hacks which may or may not impact the smart contract execution. Low-risk re-entrancy-related vulnerabilities should be fixed to deter exploits.
Minor $	These risks do not pose a considerable risk to the contract or those who interact with it. They are code-style violations and deviations from standard practices. They should be highlighted and fixed nonetheless.
Unknown %	These risks pose uncertain severity to the contract or those who interact with it. They should be fixed immediately to mitigate the risk uncertainty.

All statuses which are identified in the audit report are categorized here for the reader to review:

Status Type	Definition
Open	Risks are open.
Acknowledged	Risks are acknowledged, but not fixed.
Resolved	Risks are acknowledged and fixed.

CENTRALIZED PRIVILEGES

Centralization risk is the most common cause of cryptography asset loss. When a smart contract has a privileged role, the risk related to centralization is elevated.

There are some well-intended reasons have privileged roles, such as:

- **Privileged roles can be granted the power to** pause() **the contract in case of an external attack.**
- **Privileged roles can use functions like,** include(), **and** exclude() **to add or remove wallets from fees, swap checks, and transaction limits. This is useful to run a presale and to list on an exchange.**

Authorizing privileged roles to externally-owned-account (EOA) is dangerous. Lately, centralization-related losses are increasing in frequency and magnitude.

- **The client can lower centralization-related risks by implementing below mentioned practices:**
- **Privileged role's private key must be carefully secured to avoid any potential hack.**
- **Privileged role should be shared by multi-signature (multi-sig) wallets.**
- **Authorized privilege can be locked in a contract, user voting, or community DAO can be introduced to unlock the privilege.**
- **Renouncing the contract ownership, and privileged roles.**
- **Remove functions with elevated centralization risk.**

ℹ️ **Understand the project's initial asset distribution. Assets in the liquidity pair should be locked. Assets outside the liquidity pair should be locked with a release schedule.**

AUTOMATED ANALYSIS

Symbol	Definition
🛑	**Function modifies state**
#️⃣	**Function is payable**
🔒	**Function is internal**
🔐	**Function is private**
❗	**Function is important**

MOVE	Interface			
└	stake	External 🔒	❗	NO🔒
└	decimals	External 🔒	❗	NO🔒
└	symbol	External 🔒	❗	NO🔒
└	name	External 🔒	❗	NO🔒
└	getOwner	External 🔒		NO🔒
└	reward time	External 🔒	❗	NO❗
└	borrow	External 🔒	" ❗ 🛑	NO❗
└	allowance	External 🔒	❗	NO❗
└	approve	External 🔒	" ❗ 🛑	NO❗
└	withdraw	External 🔒	" ❗ 🛑	NO❗

IFactoryV2	Interface			
└	getPair	External 🔒	❗	NO❗
└	createPair	External 🔒	" ❗ 🛑	NO❗

IV2Pair	Interface			
└	factory	External 🔒	❗	NO❗
└	getReserves	External 🔒	❗	NO❗
└	sync	External 🔒	" ❗ 🛑	NO❗

IRouter01	Interface				
└	factory	External	!		NO!
└	APT	External	!		NO!
└	addLiquidityAPT	External	!	#$	NO!
└	addLiquidity	External	!	🛑	NO!
└	swapExactAPTForTokens	External	!	#$	NO!
└	getAmountsOut	External	!		NO!
└	getAmountsIn	External	!		NO!
IRouter02	Interface	IRouter01			
└	swapExactTokensForAPTSupportingFeeOnTransferTokens	External	!	🛑	NO!
└	swapExactAPTForTokensSupportingFeeOnTransferTokens	External	!	#$	NO!
└	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	!	🛑	NO!
└	swapExactTokensForTokens	External	!	🛑	NO!
Protections	Interface				
└	checkUser	External	!	🛑	NO!
└	setLaunch	External		🛑	NO!
└	setLpPair	External		🛑	NO!
└	setProtections	External	!	🛑	NO!
└	removeSniper	External	!	🛑	NO!
Cashier	Interface				
└	setRewardsProperties	External	!	🛑	NO!
└	tally	External	!	🛑	NO!
└	load	External	!	#$	NO!
└	cashout	External	!	🛑	NO!
└	giveMeWelfarePlease	External	!	🛑	NO!
└	getTotalDistributed	External	!		NO!
└	getUserInfo	External	!		NO!
└	getUserRealizedRewards	External	!		NO!

| └ | getPendingRewards | External ❗ | ❗ | NO❗ |

| └ | initialize | External ❗ | " ❗ 🛑 | NO❗ |

| └ | getCurrentReward | External ❗ | ❗ | NO❗ |

||||||

| **MOVE** | Implementation | **toml** |||

| └ | <Constructor> | Public ❗ | ❗ | #🆂 | NO❗| |

| └ | transferOwner | External ❗ | " ❗ 🛑 | onlyOwner |

| └ | renounceOwnership | External ❗ | " ❗ 🛑 | NO❗|

| └ | setOperator | Public ❗ | " ❗ 🛑 | NO❗ |

| └ | renounceOriginalDeployer | External ❗ | " ❗ 🛑 | NO❗ |

| └ | <Receive Ether> | External ❗ | ❗ | #🆂 | NO❗| |

| └ | totalSupply | External ❗ | ❗ | NO❗ |

| └ | decimals | External ❗ | ❗ | NO❗ |

| └ | symbol | External ❗ | ❗ | NO❗ |

| └ | name | External ❗ | ❗ | NO❗ |

| └ | getOwner | External ❗ | ❗ | NO❗ |

| └ | balanceOf | Public ❗ | ❗ | NO❗ |

| └ | allowance | External ❗ | ❗ | NO❗ |

| └ | approve | External ❗ | " ❗ 🛑 | NO❗ |

| └ | _approve | Internal $ | "🔒 🛑 | |

| └ | approveContractContingency | Public ❗ | " ❗ 🛑 | onlyOwner |

| └ | transfer | External ❗ | " ❗ 🛑 | NO❗ |

| └ | transferFrom | External ❗ | " ❗ 🛑 | NO❗ |

| └ | setNewRouter | External ❗ | " ❗ 🛑 | onlyOwner |

| └ | setLpPair | External ❗ | " ❗ 🛑 | onlyOwner |

| └ | setInitializers | External ❗ | " ❗ 🛑 | onlyOwner |

| └ | isExcludedFromFees | External ❗ | ❗ | NO❗ |

| └ | isExcludedFromDividends | External ❗ | ❗ | NO❗ |

| └ | isExcludedFromProtection | External ❗ | ❗ | NO❗ |

| └ | setDividendExcluded | Public ❗ |" 🛑 | onlyOwner |

| └ | setExcludedFromFees | Public ❗ |" 🛑 | onlyOwner |

└	setExcludedFromProtection	External		onlyOwner
└	removeSniper	External		onlyOwner
└	setProtectionSettings	External		onlyOwner
└	setWallets	External		onlyOwner
└	lockTaxes	External		onlyOwner
└	setTaxes	External		onlyOwner
└	setRatios	External		onlyOwner
└	getTokenAmountAtPriceImpact	External			NO
└	setSwapSettings	External		onlyOwner
└	setPriceImpactSwapAmount	External		onlyOwner
└	setContractSwapEnabled	External		onlyOwner
└	setRewardsProperties	External		onlyOwner
└	setReflectorSettings	External		onlyOwner
└	excludePresaleAddresses	External		onlyOwner

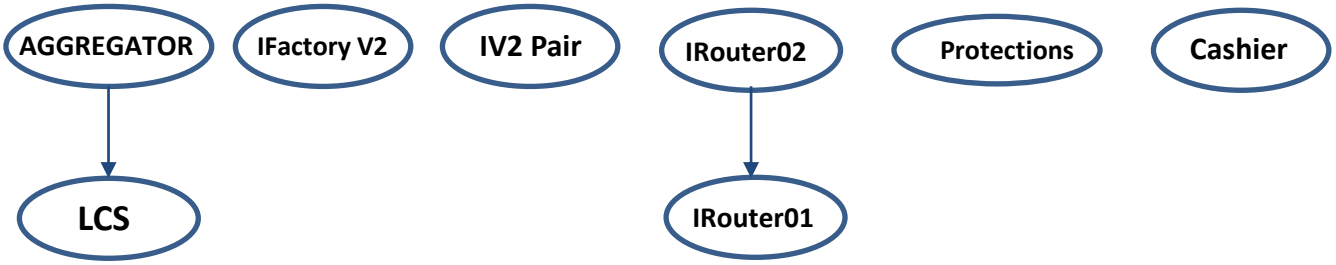
└	contractSwap	Internal		inSwapFlag
└	_checkLiquidityAdd	Private			
└	enableTrading	Public		onlyOwner
└	finalizeTransfer	Internal			
└	processRewards	Internal			
└	manualProcess	External		NO
└	takeTaxes	Internal			
└	multiSendTokens	External		onlyOwner
└	manualDeposit	External			NO
└	sweepContingency	External		onlyOwner
└	sweepExternalTokens	External		onlyOwner
└	claimPendingRewards	External		NO
└	getTotalReflected	External			NO
└	getUserInfo	External			NO

| └ | getUserRealizedGains | External ❗ | ❗ | NO❗ |

| └ | getUserUnpaidEarnings | External ❗ | ❗ | NO❗ |

| └ | getCurrentReward | External ❗ | ❗ | NO❗ |

INHERITANCE GRAPH



Identifier	Definition	Severity
CEN-16	Centralization privileges of LEONICORN SWAP	Minor $

Vulnerability 0 : No important security issue detected.
Threat level: Non

AGGREGATOR.MOVE

```
305                 abort E_INVALID_PAIR_OF_DITTO
306             }
307         }
308         else if (dex_type == DEX_TORTUGA){
309             use tortuga::stake_router;
310             use TortugaGovernance::staked_aptos_coin;
311             if (
312                 type_of<X>() == type_of<AptosCoin>() &&
313                     type_of<Y>() == type_of<staked_aptos_coin::StakedAptosCoin>()){
314
315                 let tortuga_signer = account::create_signer_with_capability(
316                     &borrow_global<TortugaSigner>(@lcs_aggregator).signerCapability
317                 );
318                 // deposit to torguga signer
319                 let tortuga_siger_addr = address_of(&tortuga_signer);
320                 coin::deposit(address_of(&tortuga_signer),x_in);
321                 // stake use tortuga signer
322                 stake_router::stake(
323                     &tortuga_signer,
324                     coin_in_value
325                 );
326                 // withdraw from tortuga signer
327                let y_out = coin::withdraw<Y>(&tortuga_signer,coin::balance<Y>(tortuga_siger_addr));
328                 (option::none(), y_out)
329             }
330             else {
```

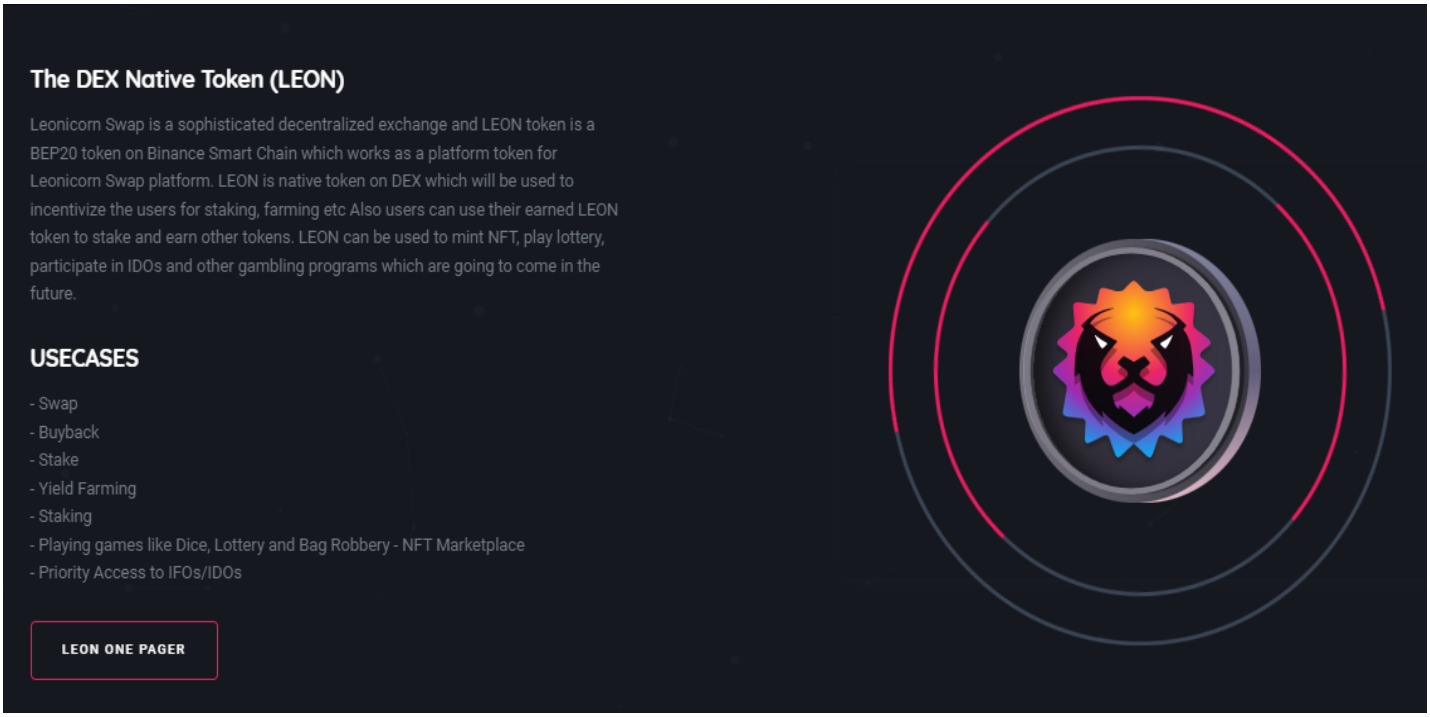
MANUAL REVIEW

LEONICORN SWAP: is an advanced Automated Market Maker AMM based, dual token model, decentralised exchange (DEX) with multifunctional features. Our aim is to deliver the next generation of efficient DeFi solutions built on blockchain technology. A user-friendly and security focused DeFi solution built for, and run for, you - with access to unparalleled live support, and vibrant online communities.

CONTRACT NAME: AGGREGATOR
LOCKING: TOKENS
TICKER: LCS

CHAIN/STANDERD: APTOS BLOCKCHIN

Outstanding features Of LEONICORN SWAP Protocol Overview



ISSUES CHECKING STATUS

	Issue Description	Checking Status
1.	Compiler errors.	PASSED
2.	Race Conditions and reentrancy. Cross-Function Race Conditions.	PASSED
3.	Possible Delay In Data Delivery.	PASSED
4.	Oracle calls.	PASSED
5.	Front Running.	PASSED
6.	Move Dependency.	PASSED
7.	Integer Overflow And Underflow.	PASSED
8.	DoS with Revert.	PASSED
9.	Dos With Block Gas Limit.	PASSED
10.	Methods execution permissions.	PASSED
11.	Economy Model of the contract.	PASSED
12.	The Impact Of Exchange Rate On the Move Logic.	PASSED
13.	Private use data leaks.	PASSED
14.	Malicious Event log.	PASSED
15.	Scoping and Declarations.	PASSED
16.	Uninitialized storage pointers.	PASSED
17.	Arithmetic accuracy.	PASSED
18.	Design Logic.	PASSED
19.	Cross-Function race Conditions	PASSED
20.	Save Upon Move contract Implementation and Usage.	PASSED
21.	Fallback Function Security	PASSED

AUDIT RESULT

PASSED

Identifier	Definition	Severity
CEN-02	Initial asset distribution	Minor

All of the initially pool assets are sent to have key contract deployer when staking funds. This Is a secured minor as the staked asset and/or contract owner can distribute tokens safely on the network with entry delegations.

```
assert!(fee_bips < 30, E_FEE_BIPS_TO_LARGE);
        let fee_value = coin::value(coin) * (fee_bips as u64) / 10000;
        if (coin::is_account_registered<X>(fee_to)){
            if (coin::is_account_registered<X>(@lcs_aggregator)){
                coin::deposit(fee_to, coin::extract(coin, fee_value/2));
                coin::deposit(@lcs_aggregator, coin::extract(coin, fee_value/2));
            } else {
                coin::deposit(fee_to, coin::extract(coin, fee_value))
```

RECOMMENDATION

Project stakeholders should be consulted during the initial asset distribution process.

RECOMMENDATION

Deployer and/or contract owner private keys are secured carefully.

Please refer to PAGE-09 CENTRALIZED PRIVILEGES for a detailed understanding.

ALLEVIATION

LEONICORN SWAP project team understands the centralization risk. Some functions are provided privileged access to ensure a good runtime behaviour in the project

Identifier	Definition	Severity
COD-18	Third Party Dependencies	Minor

Smart contract is interacting with third party protocols e.g., Pancakeswap router, cashier contract, protections contract. The scope of the audit treats third party entities as black boxes and assumes their functional correctness. However, in the real world, third parties can be compromised, and exploited. Moreover, upgrades in third parties can create severe impacts, e.g., increased transactional fees, deprecation of previous routers, etc.

RECOMMENDATION

Inspect and validate third party dependencies regularly, and mitigate severe impacts whenever necessary.

DISCLAIMERS

Vital Block provides the easy-to-understand audit of Solidity, Move and Raw source codes (commonly known as smart contracts).

The smart contract for this particular audit was analyzed for common contract vulnerabilities, and centralization exploits. This audit report makes no statements or warranties on the security of the code. This audit report does not provide any warranty or guarantee regarding the absolute bug-free nature of the smart contract analyzed, nor do they provide any indication of the client's business, business model or legal compliance. This audit report does not extend to the compiler layer, any other areas beyond the programming language, or other programming aspects that could present security risks. Cryptographic tokens are emergent technologies, they carry high levels of technical risks and uncertainty. You agree that your access and/or use, including but not limited to any services, reports, and materials, will be at your sole risk on an as-is, where-is, and as-available basis. This audit report could include false positives, false negatives, and other unpredictable results.

CONFIDENTIALITY

This report is subject to the terms and conditions (including without limitations, description of services, confidentiality, disclaimer and limitation of liability) outlined in the scope of the audit provided to the client. This report should not be transmitted, disclosed, referred to, or relied upon by any individual for any purpose without Vital Block prior written consent.

NO FINANCIAL ADVICE

This audit report does not indicate the endorsement of any particular project or team, nor guarantees its security. No third party should rely on the reports in any way, including to make any decisions to buy or sell a product, service or any other asset. The information provided in this report does not constitute investment advice, financial advice, trading advice, or any other sort of advice and you should not treat any of the report's content as such. This audit report should not be used in any way

to make decisions around investment or involvement. This report in no way provides investment advice, nor should be leveraged as investment advice of any sort.

FOR AVOIDANCE OF DOUBT, SERVICES, INCLUDING ANY ASSOCIATED AUDIT REPORTS OR MATERIALS, SHALL NOT BE CONSIDERED OR RELIED UPON AS ANY FORM OF FINANCIAL, TAX, LEGAL, REGULATORY, OR OTHER ADVICE.

TECHNICAL DISCLAIMER

ALL SERVICES, AUDIT REPORTS, SMART CONTRACT AUDITS, OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF ARE PROVIDED "AS IS" AND "AS AVAILABLE" AND WITH ALL FAULTS AND DEFECTS WITHOUT WARRANTY OF ANY KIND. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAW, VITAL BLOCK HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO SERVICES, AUDIT REPORT, OR OTHER MATERIALS. WITHOUT LIMITING THE FOREGOING, VITAL BLOCK SPECIFICALLY DISCLAIMS ALL IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AND ALL WARRANTIES ARISING FROM THE COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

WITHOUT LIMITING THE FOREGOING, VITAL BLOCK MAKES NO WARRANTY OF ANY KIND THAT ALL SERVICES, AUDIT REPORTS, SMART CONTRACT AUDITS, OR OTHER MATERIALS, OR ANY PRODUCTS OR RESULTS OF THE USE THEREOF, WILL MEET THE CLIENT'S OR ANY OTHER INDIVIDUAL'S REQUIREMENTS, ACHIEVE ANY INTENDED RESULT, BE COMPATIBLE OR WORK WITH ANY SOFTWARE, SYSTEM, OR OTHER SERVICES, OR BE SECURE, ACCURATE, COMPLETE, FREE OF HARMFUL CODE, OR ERROR-FREE.

TIMELINESS OF CONTENT

The content contained in this audit report is subject to change without any prior notice. Vital Block does not guarantee or warrant the accuracy, timeliness, or completeness of any report you access using the internet or other means, and assumes no obligation to update any information following the publication.

LINKS TO OTHER WEBSITES

This audit report provides, through hypertext or other computer links, access to websites and social accounts operated by individuals other than Vital Block. Such hyperlinks are provided for your reference and convenience only and are the exclusive responsibility of such websites and social accounts owners. You agree that Vital block Security is not responsible for the content or operation of such websites and social accounts and that Vital Block shall have no liability to you or any other person or entity for the use of third-party websites and social accounts. You are solely responsible for determining the extent to which you may use any content at any other websites and social accounts to which you link from the report.

ABOUT VITAL BLOCK

Vital Block provides intelligent blockchain Security Solutions. We provide solidity and Raw Code Review, testing, and auditing services. We have Partnered with 15+ Crypto Launchpads, audited 50+ smart contracts, and analyzed 200,000+ code lines. We have worked on major public blockchains e.g., Ethereum, Binance, Cronos, Doge, Polygon, Avalanche, Metis, Fantom, Bitcoin Cash, Aptos, Oasis, etc.

Vital Block is Dedicated to Making Defi & Web3 A Safer Place. We are Powered by Security engineers, developers, UI experts, and blockchain enthusiasts. Our team currently consists of 5 core members, and 4+ casual contributors.

Website: https://Vitalblock.org

Email: info@vitalblock.org

GitHub: https://github.com/vital-block

Twitter: https://twitter.com/VB_Audit

Telegram (Engineering): https://t.me/vital_block

Telegram (Onboarding): https://t.me/vitalblock_cmo

vital-block

info@vitalblock.org

www.Vitalblock.org

Vital Block Dedicated to Securing Public and Private Blockchain Ecosystem