STAYSAFU AUDIT

SECURITY ASSESSMENT: SEPTEMBER 30TH, 2021

LEGAL LEAF

TABLE OF CONTENTS

SUMMARY

*This report has been prepared for **Legal LEAF** to discover issues and vulnerabilities in the source code of the **Legal LEAF** project as well as any contract dependencies that were not part of an officially recognized library.*

*A comprehensive examination has been performed, utilizing Static Analysis, Manual Review, and **TESTNET** Deployment techniques. The auditing process pays special attention to the following considerations:*

- Testing the smart contracts against both common and uncommon attack vectors
- Assessing the codebase to ensure compliance with current best practices and industry standards
- Ensuring contract logic meets the specifications and intentions of the client
- Cross referencing contract structure and implementation against similar smart contracts produced by industry leaders
- Thorough line-by-line manual review of the entire codebase by industry experts

The security assessment resulted in findings that ranged from critical to informational. We recommend addressing these findings to ensure a high level of security standards and industry practices.

We suggest recommendations that could better serve the project from the security perspective :

- Enhance general coding practices for better structures of source codes
- Add enough unit tests to cover the possible use cases
- Provide more comments per each function for readability, especially contracts that are verified in public
- Provide more transparency on privileged activities once the protocol is live

OVERVIEW

VULNERABILITY SUMMARY

UNDERSTANDING

The **Legal LEAF** Protocol is a decentralized finance (**DeFi**) token deployed on the Binance smart chain (**BSC**)

Legal LEAF employs two features in its protocol static rewards for each user as well as an LP acquisition mechanism. The static rewards (also known as reflection) and LP acquisition mechanisms function as follows :

Each **Legal LEAF** transaction is taxed two **5%** fees totaling 10% of the transaction amount. The first fee is redistributed to all existing holders using a form of rebasing mechanism whilst the other **5%** is accumulated internally until a sufficient amount of capital has been amassed to perform an **LP** acquisition. When this number is reached, the total tokens accumulated are split with half being converted to **BNB** and the total being supplied to the **PANCAKESWAP** contract as liquidity.

PRIVILEGED FUNCTIONS

The contract contains the following privileged functions that are restricted by the onlyOwner modifier.
They are used to modify the contract configurations and address attributes. We grouped these functions below :

OWNERSHIP MANAGEMENT

- renounceOwnership ()
- transferOwnership (address newOwn)
- lock (uint256 time)

ACCOUNTS MANAGEMENT

- excludeFromReward (address account)
- includeInReward (address account)
- excludeFromFee (address account)
- includeInFee (address account)

TAXES MANAGEMENT

- SetTaxFeePercent (uint256 taxFee)
- includeInFee (address account)

LIQUIDITY MANAGEMENT

- SetNumTokensSellToAddToLiquidity ()
- SetSwapAndLiquifyEnabled()

TRANSACTION MANAGEMENT

- SetMaxTxPercent (uint256 maxTxPercent)

FINDINGS

Centralized risk in addLiquidity

The addLiquidity function calls the UniswapV2Router.addLiquidityETH function with the to address specified as owner() for acquiring the generated LP tokens from the Legal **LEAF/BNB** pool.

As a result, over time the _owner address will accumulate a significant portion of **LP** tokens. If the _owner is an EOA (Externally Owned Account) mishandling of its private key can have devastating consequences to the project as a whole.

We advise the to address of the UniswapV2Router.addLiquidityETH function call to be replaced by the contract itself, i.e. address (this) and to restrict the management of the LP tokens within the scope of the contract's business logic.

This will also protect the LP tokens from being stolen if the _owner account is compromised. In general, we strongly recommend centralized privileges or roles in the protocol to be improved via a decentralized mechanism or via smart-contract based accounts with enhanced security practices f.e. multi signature wallets.

The owner of contract Legal LEAF has no permission to change the contract in any way (ownership renounced)

Renounce ownership when it is the right timing, or gradually migrate to a timelock plus multisig governing procedure and let the community monitor in respect of transparency considerations.

Incorrect error message

The error message in require(_isExcluded[account]"Account is already excluded") does not describe the error correctly.

The message "Account is already excluded" can be changed to "Account is not excluded ».

Third-party dependencies

The contract is serving as the underlying entity to interact with third party **PANCAKESWAP** protocols. The scope of the audit would treat those third party entities as black boxes and assume it's functional correctness. However in the real world,

third parties may be compromised that led to assets lost or stolen.

We understand that the business logic of the **Legal LEAF** protocol requires the interaction PancakeSwap protocol for adding liquidity to Legal **LEAF/BNB** pool and swap tokens. We encourage the team to constantly monitor the statuses of those third parties to mitigate the side effects when unexpected activities are observed.

DISCLAIMER

This report is subject to the terms and conditions (including without limitation, description of services, confidentiality, disclaimer and limitation of liability) set forth in the Services Agreement, or the scope of services, and terms and conditions provided to the Company in connection with the Agreement.

This report provided in connection with the Services set forth in the Agreement shall be used by the Company only to the extent permitted under the terms and conditions set forth in the Agreement.

This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes without **StaySAFU'**s prior written consent.This report is not, nor should be considered, an “endorsement” or “disapproval” of any particular project or team. This report is not, nor should be considered, an indication of the economics or value of any “product” or “asset” created by any team or project that contracts **StaySAFU** to perform a security assessment.
This report does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors, business, business model or legal compliance.

This report should not be used in any way to make decisions around investment or involvement with any particular project.

This report in no way provides investment advice, nor should be leveraged as investment advice of any sort. This report represents an extensive assessing process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology. Blockchain technology and cryptographic assets present a high level of ongoing risk.

StaySAFU's position is that each company and individual are responsible for their own due diligence and continuous security. **StaySAFU**'s goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies, and in no way claims any guarantee of security or functionality of the technology we agree to analyze.