INFLEX

Code Review

GEMZ "LITE" SECURITY AUDIT
PREPARED ON
5-18-2021

Table of Contents

Summary

What follows is a “lite security report” of the INFLEX project.

The information appearing in this report is for general purposes only and **is not intended to provide any legal security guarantees to any individual or entity**. It is advisable to **conduct more reviews or/and audits** as just one review or audit may miss important issues. The report **does not provide personalized investment advice or recommendations**, nor does it provide advice to perform any transactions and it **does not provide investment, financial, legal, or tax advice**. We are **not responsible or liable for any loss** which results from the report.

This report should not be considered as investment advice.

Report

INFLEX Token

This contract is a fork of RFI, with some modifications of features made.

Token Supply
The total supply of INFLEX is 1,000,000,000,000,000. The number of decimal places is 9.

Transfer Maximum Size
There is a maximum amount per transaction, set to 500,000,000,000 INFLEX. If a transfer greater than this amount is made, and the sender or recipient is not the owner address of the INFLEX contract, the transaction will revert and not successfully complete.

Transfer Fees
Besides the reflection fee, there is also a liquidity fee. 5% is reflected, 5% is converted into liquidity. The fee percentages can be changed by the owner, but is capped at a maximum of 5% respectively, and has no lower cap.

When a token transfer is done, and it is not in _isExcludedFromFee, it will do the following to deduct fees calculated:
_takeLiquidity
_reflectFee
If the to or from address for that transfer is whitelisted in _isExcludedFromFee, there will be no fees charged for that transaction.

The adding of liquidity fees to the Pancakeswap INFLEX/BNB liquidity is only done in a transfer when the INFLEX contract's balance of INFLEX is greater than or equals to 500,000,000,000 INFLEX. The amount used for the liquidity will be 500,000,000,000 INFLEX.

In the swapAndLiquify function, half of the INFLEX tokens will be swapped for BNB, and this BNB and the remaining INFLEX will then be used to provide liquidity. The LP token will be sent to the address 0x0, which is a burn address. This ensures that the provided liquidity is forever locked.

The remaining BNB in the token contract after adding liquidity will be used to buy INFLEX, which will be sent to the address 0x0, which is a burn address. This causes a deflationary supply of INFLEX over time.

Contract Ownership
After deploying the INFLEX contract, the owner will be the address that deployed the contract.

The following onlyOwner functions can be called by the owner address:
renounceOwnership()
transferOwnership(address newOwner)
excludeFromReward(address account)
includeInReward(address account)
excludeFromFee(address account)
includeInFee(address account)
setTaxFeePercent(uint256 taxFee)
setLiquidityFeePercent(uint256 liquidityFee)
setMaxTxPercent(uint256 maxTxPercent)
setNumTokensSellToAddToLiquidity(uint256 _numTokensSellToAddToLiquidity)
setSwapAndLiquifyEnabled(bool _enabled)

Token Holders
At the time of the audit, the contract has not been deployed. Thus, there is no way to determine the top holders of the INFLEX token.

Timelock
The timelock is using Compound's timelock contract. The only change made is that the minimum delay is set to 1 day, instead of 2 days in the original contract.

Issues

Rated on a scale of: Infomational, Low, Medium, High, Critical

IFX-001: Sensitive state variable modifying functions can be changed instantaneously
Severity: Medium
Status: Open

The following functions can be called to change state variables without any timelock:
excludeFromReward(address account)
includeInReward(address account)
excludeFromFee(address account)
includeInFee(address account)
setTaxFeePercent(uint256 taxFee)
setLiquidityFeePercent(uint256 liquidityFee)
setMaxTxPercent(uint256 maxTxPercent)
setNumTokensSellToAddToLiquidity(uint256 _numTokensSellToAddToLiquidity)
setSwapAndLiquifyEnabled(bool _enabled)

Each of these functions can result in behavior that differs from expectations on how the token contract should behave.

Recommendations
All the above-mentioned functions should be timelocked. Also, if it is determined that no more changes need to be made, ownership can be renounced.

Notes from the team
The ownership of the token contract will be transferred to a timelock contract after the necessary actions like providing initial liquidity are done.

IFX-002: Contract deployer is excluded from fees in the constructor
Severity: Low
Status: Open

When the contract is created, the owner address is added to the _isExcludedFromFee mapping in the constructor.

As this address is an externally owned address, the wallet owner will be able to make transfers and trade INFLEX tokens without any fees.

Recommendations:
Set the contract deployer address in _isExcludedFromFee to false.

Notes from the team
The contract deployer's address will be set into false in _isExcludedFromFee once initial liquidity has been seeded.

IFX-003: Swap Router used for swapping and adding liquidity is hardcoded
Severity: Informational
Status: Open

The router address used is hardcoded to 0x10ED43C718714eb63d5aA57B78B54704E256024E (PCS Router v2).

As observed with some other RFI tokens with similar swap and liquidity provision built into the token's features, the token will be unable to migrate it's liquidity to a newer version of the automated market maker's router contract that is newly deployed.

Recommendations:
Add a privileged function to allow updating of the router address. This, however, will add an additional risk of the privileged account changing it to a malicious address which can be used to drain tokens intended for liquidity provision.

Notes from the team
The team has acknowledged this issue, but decided to keep the router addresses immutable to ensure that no tampering can be done.

Smart Contract Risk Factors

- Users should note that fees will be deducted from every INFLEX token transfer.
- When swapping INFLEX, due to the fees, the user will have to set a slippage percentage that is higher than the total fee percentage (10%), or will not be able to successfully make the swapping.
- There could be the possibility of a huge price impact due to the sale of 250,000,000,000 INFLEX for liquidity in a single transaction if the price of INFLEX eventually grows high enough.
- Transfers could be made to fail if the owner of the contract sets certain sensitive state variables to invalid values. This is now partially mitigated with the upper and lower limits in the setter functions.
- The contract owner may abuse the ability to whitelist addresses to not be subjected to fees if not removed from the whitelist. This can still be done as the ownership has not been renounced.
- Users should monitor the top holders of the token after it has been deployed. Ideally, the top holder should be the liquidity pair token contract.

RECOMMENDATIONS

- It is recommended at the very least to make the owner a timelock contract instead of an externally owned address. This will allow for time delays before sensitive changes to the token contract are made.
- The owner of a timelock can be made to be a multisig contract such as Gnosis Safe. Key members of the community can be appointed as key holders, and changes can only be executed if the threshold is met.
- The owner on the token contract should monitor the price and change the amount required to be added to liquidity preemptively.
- The owner of the contract should proactively communicate any upcoming changes to the contract's state variables before queuing a function in the time lock.