

TRYNOS AUDITS

Binance Smart Chain:
CUPLR

[0x504ae030f0997dae6b74dc0192b0478fa2ade3d3](https://bscscan.com/address/0x504ae030f0997dae6b74dc0192b0478fa2ade3d3)

CONTACT ME ON TELEGRAM



Trynos



TrynosTokenTerminal

**THIS IS A DRAFT TEMPLATE, AND IS NOT TO BE
CONSIDERED A LIVE AUDIT OR THE WORK OF TRYNOS
DISCLAIMER**

THIS IS A SMART CONTRACT AUDIT, PAID FOR BY THE PROJECT OWNERS
I HOLD NO ASSOCIATION, AFFILIATION, OR ANY RELATION TO THE
PROJECT OR IT'S TEAM. NOTHING IN THIS REPORT IS FINANCIAL ADVICE.
PLEASE DO YOUR OWN RESEARCH.

Project Description

A simple RFI fork with an extra token tax for an external owned wallet. Nothing here is of note, though there are patches of un-optimized code as well as meaningless functions that are never called.

Developer Provided Description

Social media app with in App purchases and coffee franchise supported by \$CPLR.



Project Information

Name

CUPLR

Launch Type

Presale,
Pinksale Launchpad

Telegram

<https://t.me/cuplrqa>

Website

<http://www.cuplr.com>

Taxes

5% Native Reflection
3% Burn
2% Operations

Token Distribution

50% Locked Tokens**
38% Liquidity
12% Presale

** Timelocks are not inspected
in this audit. Please inspect all
timelocks manually.

Launch Date

Presale:

TBA

PancakeSwap:

TBA



Security Quick Report

Contract Analyzed [Solidity v. 0.6.12]

<https://bscscan.com/address/0x504ae030f0997dae6b74dc0192b0478fa2ade3d3#code>

HIGH – 1 Risk

Medium – 0 Risks

Low – 4 Risks

Minimal – 0 Risks

Risk assessment is color-coded for convenience.

The risks are as follows, and their respective colors:

HIGH – SEVERE risk. Could either honeypot, rug, somehow scam the user, or prevent the contract from functioning as intended.

Medium – Could allow the owner to stop trading while being able to trade himself, make the user lose money, or something similar.

Low – Risks that could allow the owner to somehow have an edge, allow users he choose to have an edge, or the ability to manipulate the contract, prices, or something similar.

Minimal - Risks that are not potentially investment-threatening, but needs be mentioned anyway.



Security and Code

No Maximums on Taxes [HIGH Risk]

The owner can set taxes to any number. There are no limits, and even taxes of 100% is possible. These numbers can be manipulated freely. The owner (and anyone else set to be excluded) is exempt from taxes, and even if set to 100% would be able to sell without fees, while others would sell for no BNB.

LP Pair Not Excluded [Low Risk]

At time of audit the LP pair is not excluded from rewards. This means every single transaction, a % of the reflection rewards meant for the users instead gets sent to the LP Pair. Since the LP pair tends to have the most tokens at the start of the token, this means it gets most of the reflection reward. **Because of this, every single transaction drops the price of the token as the Pair gains tokens from reflection.**

Limitless Exclusions [Low Risk]

The owner is able to set any address to be excluded from fees or rewards at any time.

Improper Tax Code Usage [Low Risk]

Due to the taxes being stored in two separate variables, if a normal trade occurs, the taxes are set to 0, and a fee excluded user trades, the taxes will revert back to their previous values. In a situation where taxes are set to 0 and ownership is renounced, this will forever revert taxes to their previous values.



Security and Code (cont.)

Poor Optimization [Low Risk]

As this is basically a safemoon fork almost 1:1, the contract is not well optimized. There are four different transfer functions which get called based on exclusions, and two of them are redundant. There is code that takes up extra memory when the math can simply be done internally. The Deliver() function holds no purpose or use. The solidity version is not up to date and is missing critical bugfixes and changes in code that come with version 0.8, including built-in SafeMath. Taxes are calculated in their own separate functions, and getRValues() and getTValues() are poorly written and have overlapping assignments.

Most of these results cause the gas cost to rise needlessly, and the user bears the burden of paying for these gas costs.

Although the functions and methods exist for Automatic Liquidity, there are no fees that go to the contract, and thus the contract will never use these functions. Their checks and inclusion are just wasted gas.



Closing Notes

Most of the high-risk issues in this contract are limitless settable functions that the owner can act on. These are not by themselves malicious but can be abused with poor ownership. Please do your own research and ensure you have trust in the project and its team before investing.

Always make sure to always inspect all values and variables.

This includes, but is not limited to:

- Ownership
- Taxes
- Transaction/Wallet Limits
- Token Distributions
- Timelocks
- Liquidity Locks
- Any other owner-adjustable settings or variables.



Final Disclaimer

I am solely a smart contract writer/auditor. I do not promote, own, or lead projects unless specifically and explicitly stated. I am not a good trader, I know nothing of Technical Analysis, and for the most part I do not trade. Despite my best efforts, I can never guarantee full safety, though I do my best to check for and announce all risks. There are many tricks and variations that unscrupulous people can enact even with safe contracts. Please always do your research.

I am only human, and I make mistakes as well.

Nothing written in this report is financial advice. Please make sure you are interacting with the correct socials below when contacting me, to avoid scamming impersonators.

CONTACT ME ON [TELEGRAM](#)



@Trynos



TrynosTokenTerminal