



TechRate

AUDIT COMPANY

Smart Contract Security Audit

TechRate

June, 2021

Audit Details



Audited project

HyperDeflate



Deployer address

0x7b46B37ccF5407e24f32e16551648e9EcA4CBBed



Client contacts:

HyperDeflate team



Blockchain

Binance Smart Chain



Project website:

<https://www.HyperDeflate.com>

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the below disclaimer below – please make sure to read it in full.

DISCLAIMER: By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice. No one shall have any right to rely on the report or its contents, and TechRate and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (TechRate) owe no duty of care towards you or any other person, nor does TechRate make any warranty or representation to any person on the accuracy or completeness of the report. The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and TechRate hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, TechRate hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against TechRate, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report.

The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.

Background

TechRate was commissioned by HyperDeflate to perform an audit of smart contracts:

<https://bscscan.com/address/0x0b4922132ec425806fd8aee94b5788f9b70d60ca#code>

The purpose of the audit was to achieve the following:

- Ensure that the smart contract functions as intended.
- Identify potential security issues with the smart contract.

The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

Contracts Details

Token contract details for 20.06.2021

Contract name	HyperDeflate
Contract address	0x0b4922132ec425806Fd8AEE94b5788F9B70D60ca
Total supply	1,000,000,000,000,000
Token ticker	HDFL
Decimals	18
Token holders	4,434
Transactions count	9,956
Top 100 holders dominance	99.29%
Liquidity fee	0
Tax fee	0
Total fees	0
Uniswap V2 pair	0xba3cfd32ce3948584223a44b65e965d8c4711865
Contract deployer address	0x7b46B37ccF5407e24f32e16551648e9EcA4CBBed
Contract's current owner address	0x00

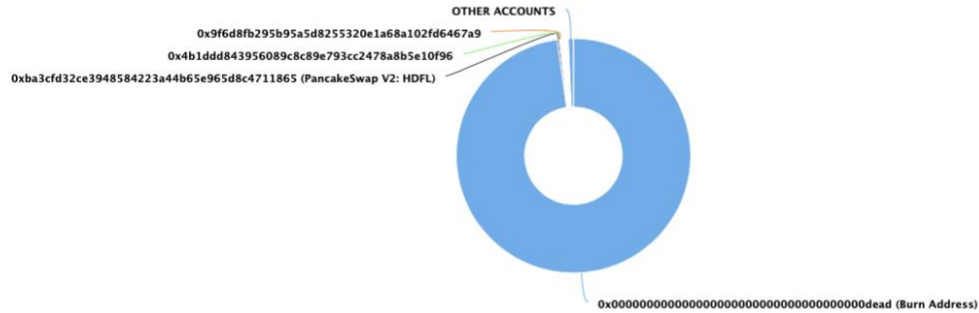
HyperDeflate Token Distribution

The top 100 holders collectively own 99.29% (992,877,758,031,771.00 Tokens) of Hyper Deflate

Token Total Supply: 1,000,000,000,000.00 Token | Total Token Holders: 4,434

Hyper Deflate Top 100 Token Holders

Source: BscScan.com



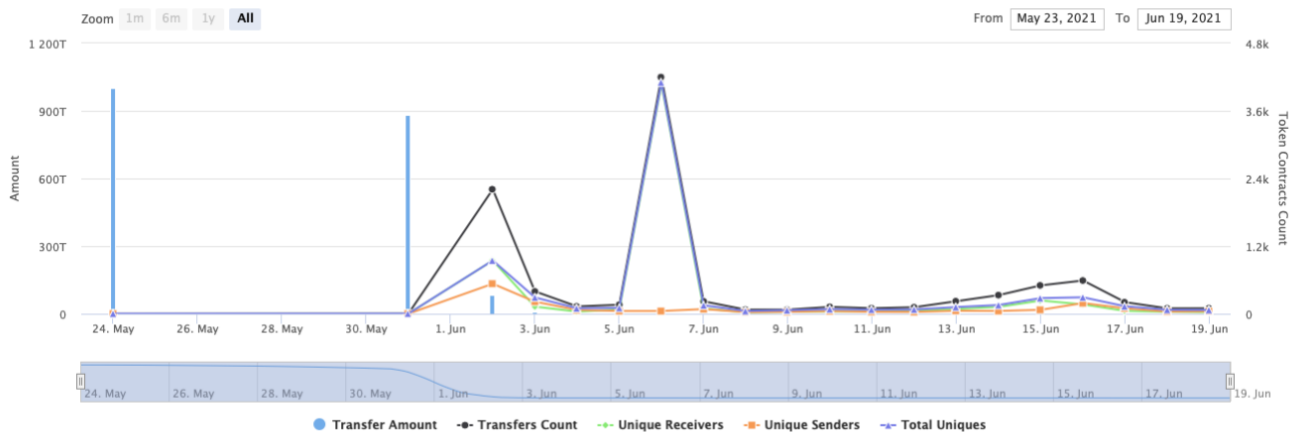
(A total of 992,877,758,031,771.00 tokens held by the top 100 accounts from the total supply of 1,000,000,000,000.00 token)

HyperDeflate Contract Interaction Details

Time Series: Token Contract Overview

Mon 24, May 2021 - Sat 19, Jun 2021

Token Contract 0x0b4922132ec425806fd8aee94b5788f9b70d60ca (Hyper Deflate)
Source: BscScan.com



HyperDeflate Top 10 Token Holders

Rank	Address	Quantity (Token)	Percentage
1	Burn Address	978,045,264,290,011.852652047412837287	97.8045%
2	PancakeSwap V2: HDFL	2,691,888,472,623.157455318237416778	0.2692%
3	0x4b1ddd843956089c8c89e793cc2478a8b5e10f96	1,207,546,949,991.952481263747686375	0.1208%
4	0x9f6d8fb295b95a5d8255320e1a68a102fd6467a9	785,992,538,961.357501048856892227	0.0786%
5	0xf0879ad8134dee6e58b2c1a7d2d29f76ab927e30	650,000,000,000	0.0650%
6	0xa589f94475a1fea52da23cc7ebadb1e80c8ef277	641,199,819,256.254568732563285473	0.0641%
7	0xb2f7a5bb8e2475ece15c2e3a2bce9159d9ee3228	502,128,034,995.725596417917471272	0.0502%
8	0xea865ff603bdc7eda3cdf6e678ac0e199ffe87a	292,490,524,601.326301359475895699	0.0292%
9	0x37862816ac973c55ffb6df21202bdebe06ad1c5	288,945,705,571.908690960330721217	0.0289%
10	0xfd88b2e7333ec2cbe17567cc210628e7aed4a1e4	276,363,886,652.644568553608459636	0.0276%

HyperDeflate LP Token Holders

Rank	Address	Quantity	Percentage
1	0x495d6f360c5a6b5681d73714a87b6c226ea6ad6	33,201,096.675491238202752672	<u>100.0000%</u>
2	0x00	0.000000000000001	<u>0.0000%</u>



Contract functions details

- + [Int] IERC20
 - [Ext] totalSupply
 - [Ext] balanceOf
 - [Ext] transfer #
 - [Ext] allowance
 - [Ext] approve #
 - [Ext] transferFrom #
- + [Lib] SafeMath
 - [Int] tryAdd
 - [Int] trySub
 - [Int] tryMul
 - [Int] tryDiv
 - [Int] tryMod
 - [Int] add
 - [Int] sub
 - [Int] mul
 - [Int] div
 - [Int] mod
 - [Int] sub
 - [Int] div
 - [Int] mod
- + Context
 - [Int] _msgSender
 - [Int] _msgData
- + [Lib] Address
 - [Int] isContract
 - [Int] sendValue #
 - [Int] functionCall #
 - [Int] functionCall #
 - [Int] functionCallWithValue #
 - [Int] functionCallWithValue #
 - [Int] functionStaticCall
 - [Int] functionStaticCall
 - [Int] functionDelegateCall #
 - [Int] functionDelegateCall #
 - [Prv] _verifyCallResult
- + Ownable (Context)
 - [Pub] <Constructor> #
 - [Pub] owner
 - [Pub] renounceOwnership #
 - modifiers: onlyOwner
 - [Pub] transferOwnership #
 - modifiers: onlyOwner
- + [Int] IUniswapV2Factory
 - [Ext] feeTo
 - [Ext] feeToSetter

- [Ext] getPair
- [Ext] allPairs
- [Ext] allPairsLength
- [Ext] createPair #
- [Ext] setFeeTo #
- [Ext] setFeeToSetter #

- + [Int] IUniswapV2Pair
 - [Ext] name
 - [Ext] symbol
 - [Ext] decimals
 - [Ext] totalSupply
 - [Ext] balanceOf
 - [Ext] allowance
 - [Ext] approve #
 - [Ext] transfer #
 - [Ext] transferFrom #
 - [Ext] DOMAIN_SEPARATOR
 - [Ext] PERMIT_TYPEHASH
 - [Ext] nonces
 - [Ext] permit #
 - [Ext] MINIMUM_LIQUIDITY
 - [Ext] factory
 - [Ext] token0
 - [Ext] token1
 - [Ext] getReserves
 - [Ext] price0CumulativeLast
 - [Ext] price1CumulativeLast
 - [Ext] kLast
 - [Ext] mint #
 - [Ext] burn #
 - [Ext] swap #
 - [Ext] skim #
 - [Ext] sync #
 - [Ext] initialize #

- + [Int] IUniswapV2Router01
 - [Ext] factory
 - [Ext] WETH
 - [Ext] addLiquidity #
 - [Ext] addLiquidityETH (\$)
 - [Ext] removeLiquidity #
 - [Ext] removeLiquidityETH #
 - [Ext] removeLiquidityWithPermit #
 - [Ext] removeLiquidityETHWithPermit #
 - [Ext] swapExactTokensForTokens #
 - [Ext] swapTokensForExactTokens #
 - [Ext] swapExactETHForTokens (\$)
 - [Ext] swapTokensForExactETH #
 - [Ext] swapExactTokensForETH #
 - [Ext] swapETHForExactTokens (\$)
 - [Ext] quote
 - [Ext] getAmountOut
 - [Ext] getAmountIn
 - [Ext] getAmountsOut

- [Ext] getAmountsIn
- + [Int] IUniswapV2Router02 (IUniswapV2Router01)
 - [Ext] removeLiquidityETHSupportingFeeOnTransferTokens #
 - [Ext] removeLiquidityETHWithPermitSupportingFeeOnTransferTokens #
 - [Ext] swapExactTokensForTokensSupportingFeeOnTransferTokens #
 - [Ext] swapExactETHForTokensSupportingFeeOnTransferTokens (\$)
 - [Ext] swapExactTokensForETHSupportingFeeOnTransferTokens #
- + hyperdeflate (Context, IERC20, Ownable)
 - [Pub] <Constructor> #
 - [Pub] name
 - [Pub] symbol
 - [Pub] decimals
 - [Pub] totalSupply
 - [Pub] balanceOf
 - [Pub] transfer #
 - [Pub] allowance
 - [Pub] approve #
 - [Pub] transferFrom #
 - [Pub] increaseAllowance #
 - [Pub] decreaseAllowance #
 - [Pub] isExcludedFromReward
 - [Pub] totalFees
 - [Pub] deliver #
 - [Pub] reflectionFromToken
 - [Pub] tokenFromReflection
 - [Pub] excludeFromReward #
 - modifiers: onlyOwner
 - [Ext] includeInReward #
 - modifiers: onlyOwner
 - [Prv] _transferBothExcluded #
 - [Pub] excludeFromFee #
 - modifiers: onlyOwner
 - [Pub] includeInFee #
 - modifiers: onlyOwner
 - [Ext] setTaxFeePercent #
 - modifiers: onlyOwner
 - [Ext] setCharityFeePercent #
 - modifiers: onlyOwner
 - [Ext] setLiquidityFeePercent #
 - modifiers: onlyOwner
 - [Ext] setMaxTxPercent #
 - modifiers: onlyOwner
 - [Pub] setSwapAndLiquifyEnabled #
 - modifiers: onlyOwner
 - [Ext] <Fallback> (\$)
 - [Prv] _reflectFee #
 - [Prv] _getValues
 - [Prv] _getTValues
 - [Prv] _getRValues
 - [Prv] _getRate
 - [Prv] _getCurrentSupply
 - [Prv] _takeLiquidity #
 - [Prv] _takeCharity #

- [Prv] calculateTaxFee
- [Prv] calculateCharityFee
- [Prv] calculateLiquidityFee
- [Prv] removeAllFee #
- [Prv] restoreAllFee #
- [Pub] isExcludedFromFee
- [Prv] _approve #
- [Prv] _transfer #
- [Prv] swapAndLiquify #
 - modifiers: lockTheSwap
- [Prv] swapTokensForEth #
- [Prv] addLiquidity #
- [Prv] _tokenTransfer #
- [Prv] _transferStandard #
- [Prv] _transferToExcluded #
- [Prv] _transferFromExcluded #

(\$) = payable function

= non-constant function

Issues Checking Status

Issue description	Checking status
1. Compiler errors.	Passed
2. Race conditions and Reentrancy. Cross-function race conditions.	Passed
3. Possible delays in data delivery.	Passed
4. Oracle calls.	Passed
5. Front running.	Passed
6. Timestamp dependence.	Passed
7. Integer Overflow and Underflow.	Passed
8. DoS with Revert.	Passed
9. DoS with block gas limit.	Low issues
10. Methods execution permissions.	Passed
11. Economy model of the contract.	Passed
12. The impact of the exchange rate on the logic.	Passed
13. Private user 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.	Low issues
19. Cross-function race conditions.	Passed
20. Safe Open Zeppelin contracts implementation and usage.	Passed
21. Fallback function security.	Passed

Security Issues

✓ High Severity Issues

No high severity issues found.

✓ Medium Severity Issues

No medium severity issues found.

✓ Low Severity Issues

1. Out of gas

Issue:

- The function `includeInReward()` uses the loop to find and remove addresses from the `_excluded` list. Function will be aborted with `OUT_OF_GAS` exception if there will be a long excluded addresses list.

```
function includeInReward(address account↑) external onlyOwner() {
    require(!_isExcluded[account↑], "Account is already excluded");
    for (uint256 i = 0; i < _excluded.length; i++) {
        if (_excluded[i] == account↑) {
            _excluded[i] = _excluded[_excluded.length - 1];
            tOwned[account↑] = 0;
            _isExcluded[account↑] = false;
            _excluded.pop();
            break;
        }
    }
}
```

- The function `_getCurrentSupply` also uses the loop for evaluating total supply. It also could be aborted with `OUT_OF_GAS` exception if there will be a long excluded addresses list.

```
function _getCurrentSupply() private view returns (uint256, uint256) {
    uint256 rSupply = _rTotal;
    uint256 tSupply = _tTotal;
    for (uint256 i = 0; i < _excluded.length; i++) {
        if (
            _rOwned[_excluded[i]] > rSupply ||
            _tOwned[_excluded[i]] > tSupply
        ) return (_rTotal, _tTotal);
        rSupply = rSupply.sub(_rOwned[_excluded[i]]);
        tSupply = tSupply.sub(_tOwned[_excluded[i]]);
    }
    if (rSupply < _rTotal.div(_tTotal)) return (_rTotal, _tTotal);
    return (rSupply, tSupply);
}
```

2. Wrong burning

Issue:

- Charity address is 0x000000000000000000000000000000000000dEaD, so taking charity is kind of burning.
- To burn properly, use decreasing rTotal and tTotal values instead of sending tokens to zero address.

Recommendation:

Check that the excluded array length is not too big.

Conclusion

Smart contracts contain low severity issues! Liquidity pair contract's security is not checked due to out of scope.

Liquidity locking details provided by the team:

<https://dxsale.app/app/pages/defipresale?saleID=2440&chain=BSC>

TechRate note:

Please check the disclaimer above and note, the audit makes no statements or warranties on business model, investment attractiveness or code sustainability. The report is provided for the only contract mentioned in the report and does not include any other potential contracts deployed by Owner.



[Techrate1](#)



[Techrate](#)



[Techrate audits](#)