Cyberscope

Audit Report

Little Pony

April 2022

Type	BEP20
Network	BSC
Address	0x6b927b17cbcaf814e1054ba070e73a0206d75a4c
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	LittlePony
Compiler Version	v0.6.12+commit.27d51765
Optimization	200 runs
Licence	Unlicense
Explorer	https://bscscan.com/token/0x6b927b17cbcaf814e1054ba070e73a0206d75a4c
Symbol	PONY
Decimals	18
Total Supply	99,999,999,999
Domain	littleponytoken.com

Source Files

Filename	**SHA256**
contract.sol	eacf0d92656797c0ccdaf2aba26af5d60ee5024766d9fdbee3e8070fbeecb340

Audit Updates

Initial Audit	29th April 2022
Corrected	

Contract Analysis

Critical Medium Minor Pass

Severity	Code	Description
●	ST	Contract Owner is not able to stop or pause transactions
●	OCTD	Contract Owner is not able to transfer tokens from specific address
●	OTUT	Owner Transfer User's Tokens
●	ELFM	Contract Owner is not able to increase fees more than a reasonable percent (25%)
●	ULTW	Contract Owner is not able to increase the amount of liquidity taken by dev wallet more than a reasonable percent
●	MT	Contract Owner is not able to mint new tokens
●	BT	Contract Owner is not able to burn tokens from specific wallet
●	BC	Contract Owner is not able to blacklist wallets from selling

ST - Stop Transactions

Criticality	medium
Location	contract.sol#L1048

Description

The contract owner has the authority to stop transactions for all users excluding the owner. The owner may take advantage of it by setting the _maxTxAmount to zero.

```
if(from != owner() && to != owner())
            require(amount <= _maxTxAmount, "Transfer amount exceeds the
maxTxAmount.");
```

Recommendation

The contract could embody a check for not allowing setting the _maxTxAmount less than a reasonable amount. A suggested implementation could check that the minimum amount should be more than a fixed percentage of the total supply.

The team should carefully manage the private keys of the owner’s account. We strongly recommend a powerful security mechanism that will prevent a single user from accessing the contract admin functions. That risk can be prevented by temporarily locking the contract or renouncing ownership.

Contract Diagnostics

Critical Medium Minor

Severity	Code	Description
	FSA	Fixed Swap Address
	MAL	Misused Algorithmic Logic
	CO	Code Optimization
	L01	Public Function could be Declared External
	L02	State Variables could be Declared Constant
	L04	Conformance to Solidity Naming Conventions
	L07	Missing Events Arithmetic
	L09	Dead Code Elimination
	L13	Divide before Multiply Operation

FSA - Fixed Swap Address

Criticality	minor
Location	contract.sol#L740

Description

The swap address is assigned once in the constructor and it can not be changed. The decentralized swaps sometimes create a new swap version or abandon the current. A contract that cannot change the swap address may not be able to catch-up the upgrade.

```
IUniswapV2Router02 _uniswapV2Router =
IUniswapV2Router02(0x10ED43C718714eb63d5aA57B78B54704E256024E);
        // Create a uniswap pair for this new token
        uniswapV2Pair = IUniswapV2Factory(_uniswapV2Router.factory())
        .createPair(address(this), _uniswapV2Router.WETH());
        // set the rest of the contract variables
        uniswapV2Router = _uniswapV2Router;
```

Recommendation

It could be better to allow the swap address mutation in case of future swap updates.

MAL - Misused Algorithmic Logic

Criticality	critical
Location	contract.sol#L924

Description

The algorithmic flow does not follow the required business logic. In the following code segment, the `rFee` is changed (subtraction) before they are removed from the `_rTotal`. So, the `_rTotal` will be reduced by a percentage amount instead of the current total fees. As a result, the value of the `_rTotal` will be bigger than the actual total supply.

```
                uint256 _rtotransferrteam = ronePart.mul(transferTeamfee);
                uint256 _ttotransferrteam = tonePart.mul(transferTeamfee);
                rFee = rFee.sub(_rtotransferrteam);
                tFee = tFee.sub(_ttotransferrteam);

                uint256 _rtotransferrburn = ronePart.mul(burnDivisor);
                uint256 _ttotransferrburn = tonePart.mul(burnDivisor);
                rFee = rFee.sub(_rtotransferrburn);
                tFee = tFee.sub(_ttotransferrburn);

                uint256 _rtotransferrcharity = ronePart.mul(charityDivisor);
                uint256 _ttotransferrcharity = tonePart.mul(charityDivisor);
                rFee = rFee.sub(_rtotransferrcharity);
                tFee = tFee.sub(_ttotransferrcharity);

                _takefunds(teamAddress,_rtotransferrteam,_ttotransferrteam);
                _takefunds(deadAddress,_rtotransferrburn,_ttotransferrburn);

_takefunds(airdropAddress,_rtotransferrcharity,_ttotransferrcharity);
            }
            _rTotal = _rTotal.sub(rFee);
            _tFeeTotal = _tFeeTotal.add(tFee);
```

Recommendation

The algorithm should be reshaped so it will match to the business logic.

CO - Code Optimization

Criticality	minor
Location	contract.sol#L702, 911 , 1081, 1246, 1252

Description

There are code segments that could be optimized. A segment may be optimized so that it becomes a smaller size, consumes less memory, executes more rapidly, or performs fewer operations. `teamDivisor` is fixed to 0 so the calculation will always return 0.

```
uint256 public teamDivisor = 0;
```

```
uint256 _rtotransferrteam = ronePart.mul(teamDivisor);
uint256 _ttotransferrteam = tonePart.mul(teamDivisor);
rFee = rFee.sub(_rtotransferrteam);
tFee = tFee.sub(_ttotransferrteam);
```

Contract owner has the authority to manipulate the fees by calling the `setTaxFeePercent` and `setLiquidityFeePercent` functions. But, the fees are set through `_transfer` function on each transaction.

```
if(to == address(uniswapV2Pair)  ) {
            // apply sell fee
            setSellFee();
}else{
            if(_msgSender() != address(uniswapV2Pair)){
                setTransferFee();
            }else{
                removeAllFee();
            }
}
```

```
function setSellFee() internal {
        isSell = true;
        _taxFee = 9;
        _liquidityFee = 6;
    }
```


```
function setSellFee() internal {
        isSell = true;
        _taxFee = 9;
        _liquidityFee = 6;
    }
```


Recommendation

Rewrite some code segments so the runtime will be more performant.

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L433,442,754,758,762,766,775,780,784,789 and 11 more

Description

Public functions that are never called by the contract should be declared external to save gas.

```
isExcludedFromFee
setSwapAndLiquifyEnabled
includeInFee
excludeFromFee
excludeFromReward
reflectionFromToken
deliver
totalFees
isExcludedFromReward
...
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract.sol#L691,689,690,685,708,706,705,704,707,703 and 2 more

Description

Constant state variables should be declared constant to save gas.

```
transferTeamfee
teamDivisor
sellliquiduty
marketingDivisor
farmstakingDivisor
ecosystemDivisor
charityDivisor
burnDivisor
_tTotal
...
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L486,487,504,526,897,1001,1007,1220,1224,1228 and 6 more

Description

Solidity defines a naming convention that should be followed. Rule exceptions:

- Allow constant variable name/symbol/decimals to be lowercase.
- Allow _ at the beginning of the mixed_case match for private variables and unused parameters.

```
_maxTxAmount
_liquidityFee
_taxFee
_selltokenswap
_farmsstakingAddress
_teamAddress
_airdropAddress
_foundation
_marketingAddress
...
```

Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions

L07 - Missing Events Arithmetic

Criticality	minor
Location	contract.sol#L881,886,891,1239,1242

Description

Detected missing events for critical arithmetic parameters. There are functions that have no event emitted, so it is difficult to track off-chain changes.

```
_maxTxAmount = maxTxAmount
numTokensSellToAddToLiquidity = _selltokenswap
_maxTxAmount = _tTotal.mul(maxTxPercent).div(10 ** 2)
_liquidityFee = liquidityFee
_taxFee = taxFee
```

Recommendation

Emit an event for critical parameter changes.

L09 - Dead Code Elimination

Criticality	minor
Location	contract.sol#L361,321,331,346,356,268,295

Description

Functions that are not used in the contract, and make the code's size bigger.

```
sendValue
isContract
functionCallWithValue
functionCall
_functionCallWithValue
```

Recommendation

Remove unused functions.

L13 - Divide before Multiply Operation

Criticality	minor
Location	contract.sol#L905,1095 and 6 more

Description

Performing divisions before multiplications may cause lose of prediction.

```
convertbaltoone = newBalance.div(5)
onepartliqudity = contractTokenBalance.div(_liquidityFee)
tonePart = tFee.div(_taxFee)
ronePart = rFee.div(_taxFee)
...
```

Recommendation

The multiplications should be prior to the divisions.

Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
SafeMath	Library			
	add	Internal		
	sub	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	div	Internal		
	mod	Internal		
	mod	Internal		
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	

	functionCallWithValue	Internal	✓	
	_functionCallWithValue	Private	✓	
Ownable	Implementation	Context		
	<Constructor>	Internal	✓	
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
IUniswapV2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
IUniswapV2Pair	Interface			
	name	External		-
	symbol	External		-
	decimals	External		-
	totalSupply	External		-
	balanceOf	External		-
	allowance	External		-
	approve	External	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-
	DOMAIN_SEPARATOR	External		-
	PERMIT_TYPEHASH	External		-
	nonces	External		-
	permit	External	✓	-
	MINIMUM_LIQUIDITY	External		-

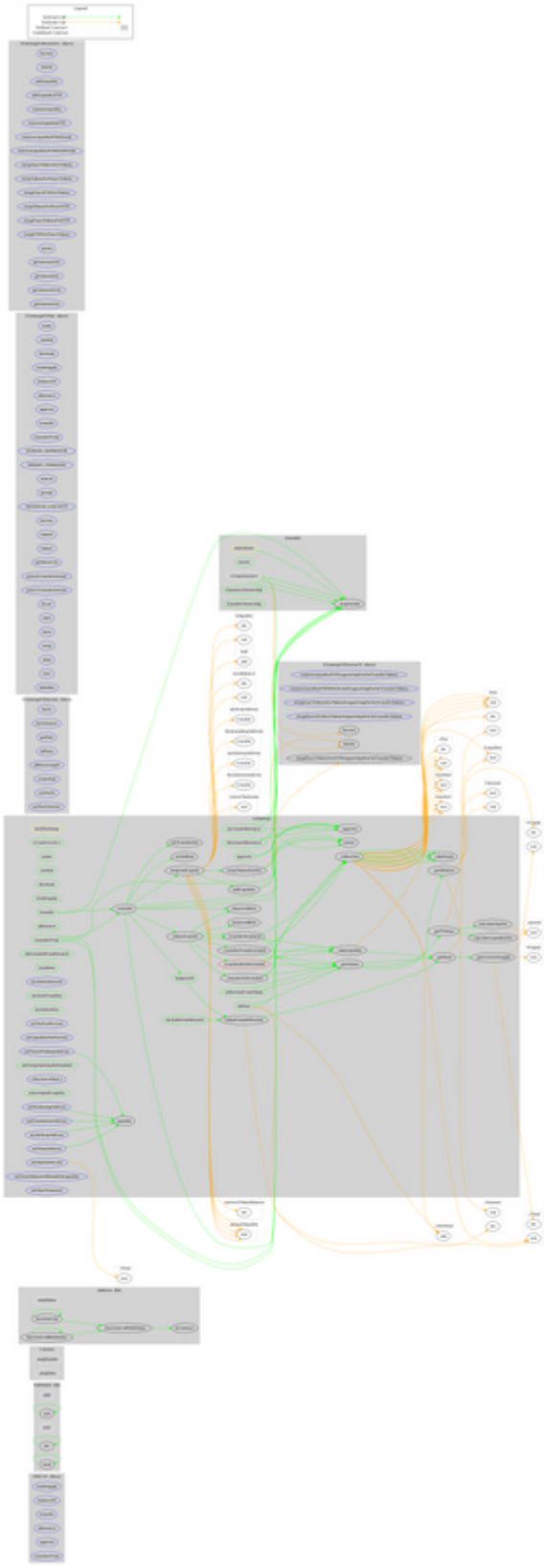
	factory	External		-
	token0	External		-
	token1	External		-
	getReserves	External		-
	price0CumulativeLast	External		-
	price1CumulativeLast	External		-
	kLast	External		-
	mint	External	✓	-
	burn	External	✓	-
	swap	External	✓	-
	skim	External	✓	-
	sync	External	✓	-
	initialize	External	✓	-
IUniswapV2Router01	Interface			
	factory	External		-
	WETH	External		-
	addLiquidity	External	✓	-
	addLiquidityETH	External	Payable	-
	removeLiquidity	External	✓	-
	removeLiquidityETH	External	✓	-
	removeLiquidityWithPermit	External	✓	-
	removeLiquidityETHWithPermit	External	✓	-
	swapExactTokensForTokens	External	✓	-
	swapTokensForExactTokens	External	✓	-
	swapExactETHForTokens	External	Payable	-
	swapTokensForExactETH	External	✓	-
	swapExactTokensForETH	External	✓	-
	swapETHForExactTokens	External	Payable	-
	quote	External		-
	getAmountOut	External		-
	getAmountIn	External		-
	getAmountsOut	External		-
	getAmountsIn	External		-

IUniswapV2Router02	Interface	IUniswapV2Router01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
LittlePony	Implementation	Context, IERC20, Ownable		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	isExcludedFromReward	Public		-
	totalFees	Public		-
	deliver	Public	✓	-
	reflectionFromToken	Public		-
	tokenFromReflection	Public		-
	excludeFromReward	Public	✓	onlyOwner
	includeInReward	External	✓	onlyOwner
	_transferBothExcluded	Private	✓	
	excludeFromFee	Public	✓	onlyOwner

	includeInFee	Public	✓	onlyOwner
	setTaxFeePercent	External	✓	onlyOwner
	setLiquidityFeePercent	External	✓	onlyOwner
	setMaxTxPercent	External	✓	onlyOwner
	setSwapAndLiquifyEnabled	Public	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	_reflectFee	Private	✓	
	_takefunds	Private	✓	
	_getValues	Private		
	_getTValues	Private		
	_getRValues	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	_takeLiquidity	Private	✓	
	calculateTaxFee	Private		
	calculateLiquidityFee	Private		
	removeAllFee	Private	✓	
	restoreAllFee	Private	✓	
	isExcludedFromFee	Public		-
	_approve	Private	✓	
	_transfer	Private	✓	
	swapAndLiquify	Private	✓	lockTheSwap
	swapTokensForEth	Private	✓	
	addLiquidity	Private	✓	
	_tokenTransfer	Private	✓	
	_transferStandard	Private	✓	
	_transferToExcluded	Private	✓	
	_transferFromExcluded	Private	✓	
	setMarketingAddress	External	✓	onlyOwner
	setFoundationAddress	External	✓	onlyOwner
	setAirdropAddress	External	✓	onlyOwner
	setTeamAddress	External	✓	onlyOwner
	setFarmsStakingAddress	External	✓	onlyOwner
	setNumTokensSellToAddToLiquidity	External	✓	onlyOwner

	setMaxTxAmount	External	✓	onlyOwner
	setSellFee	Internal	✓	
	setTransferFee	Internal	✓	

Contract Flow



Domain Info

Domain Name	littleponytoken.com
Registry Domain ID	2664486449_DOMAIN_COM-VRSN
Creation Date	2021-12-28T16:48:59Z
Updated Date	2021-12-28T16:48:59Z
Registry Expiry Date	2022-12-28T16:48:59Z
Registrar WHOIS Server	whois.name.com
Registrar URL	http://www.name.com
Registrar	Name.com, Inc.
Registrar IANA ID	625

The domain has been created 4 months before the creation of the audit. It will expire in 8 months.

There is no public billing information, the creator is protected by the privacy settings.

Summary

There are some functions that can be abused by the owner, like stopping transactions for everyone else other than the owner. The contract is also using a reflection technique that manipulates the total supply in a wrong way. The fees are fixed to 15% on selling and 5% on buying. A multi-wallet signing pattern will provide security against potential hacks. Temporarily locking the contract or renouncing ownership will eliminate all the contract threats.

Disclaimer

All the content provided in this document is for general information only and should not be used as financial advice or a reason to buy any investment.

Cyberscope team provides no guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Always Do your own research and protect yourselves from being scammed.

The Cyberscope team has audited this project for general information and only expresses their opinion based on similar projects and checks from popular diagnostic tools. Under no circumstances did Cyberscope receive a payment to manipulate those results or change the awarding badge that we will be adding in our website.

Always Do your own research and protect yourselves from scams. This document should not be presented as a reason to buy or not buy any particular token.

The Cyberscope team disclaims any liability for the resulting losses.

About Cyberscope

Coinscope audit and K.Y.C. service has been rebranded to Cyberscope.

Coinscope is the leading early coin listing, voting and auditing authority firm. The audit process is analyzing and monitoring many aspects of the project. That way, it gives the community a good sense of security using an informative report and a generic score.

Cyberscope and Coinscope are aiming to make crypto discoverable and efficient globally. They provides all the essential tools to assist users draw their own conclusions.

The Cyberscope team

https://www.cyberscope.io