



Audit Report

Apocalypse

February 2022

Type	BEP20
Network	BSC
Address	0x700AfD366D8cD7CC30f17E215F83BEeaf92459DF
Audited by	© coinscope

Table of Contents

Table of Contents	1
Contract Review	2
Audit Updates	2
Initial Audit	2
Contract Analysis	3
Contract Diagnostics	4
L01 - Public Function could be Declared External	5
Description	5
Recommendation	5
L04 - Conformance to Solidity Naming Conventions	6
Description	6
Recommendation	6
L09 - Dead Code Elimination	7
Description	7
Recommendation	7
L11 - Unnecessary Boolean equality	8
Description	8
Recommendation	8
L13 - Divide before Multiply Operation	9
Description	9
Recommendation	9
Contract Functions	10
Contract Flow	15
Summary	16
Disclaimer	17
About Coinscope	18

Contract Review

Contract Name	Apocalypse
Compiler Version	v0.8.12+commit.f00d7308
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0x700AfD366D8cD7CC30f17E215F83BEeaf92459DF
Symbol	APOC
Decimals	9
Total Supply	1,000,000,000
Source	contract.sol
Domain	

Audit Updates

Initial Audit	22nd February 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

Contract Diagnostics

● Critical ● Medium ● Minor

Severity	Code	Description
●	L01	Public Function could be Declared External
●	L04	Conformance to Solidity Naming Conventions
●	L09	Dead Code Elimination
●	L11	Unnecessary Boolean equality
●	L13	Divide before Multiply Operation

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L505,512,1184,1259,1263

Description

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

```
setRewardPoolDistributor  
distributorInitialization  
setFees  
unauthorize  
authorize
```

Recommendation

Use the external attribute for functions never called from the contract.

L04 - Conformance to Solidity Naming Conventions

Criticality

minor

Location

contract.sol#L657,805,824,832,845,857,739,740,1185,1186 and 14 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.

```
_period  
_amount  
_cap  
_enabled  
_denominator  
_target  
_marketingFeeReceiver  
_autoLiquidityReceiver  
_feeDenominator  
...
```

Recommendation

Follow the Solidity naming convention.

<https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions>

L09 - Dead Code Elimination

Criticality

minor

Location

contract.sol#L104,114,129,139,176,186,153,163,52,79 and 10 more

Description

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

```
trySub  
tryMul  
tryMod  
tryDiv  
tryAdd  
mod  
div  
_msgData  
verifyCallResult  
...
```

Recommendation

Remove unused functions.

L11 - Unnecessary Boolean equality

Criticality

minor

Location

contract.sol#L824,832,939

Description

The comparison to boolean constants is redundant. Boolean constants can be used directly and do not need to be compared to true or false.

```
require(bool,string)(buyBacker[_msgSender()] == true,Not a buybacker)  
needResetTimeLimit(_msgSender()) == true
```

Recommendation

Remove the equality to the boolean constant.

L13 - Divide before Multiply Operation

Criticality

minor

Location

contract.sol#L845,1197

Description

Performing divisions before multiplications may cause lose of prediction.

```
require(bool,string)(totalFee < feeDenominator.div(100).mul(10),Total fee  
should not be greater than 10%.)  
timeCycle = timeDifference.div(timeLimit)
```

Recommendation

The multiplications should be prior to the divisions.

Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	verifyCallResult	Internal		
SafeMath	Library			
	tryAdd	Internal		
	trySub	Internal		
	tryMul	Internal		
	tryDiv	Internal		
	tryMod	Internal		
	add	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	mod	Internal		
	sub	Internal		
	div	Internal		
	mod	Internal		

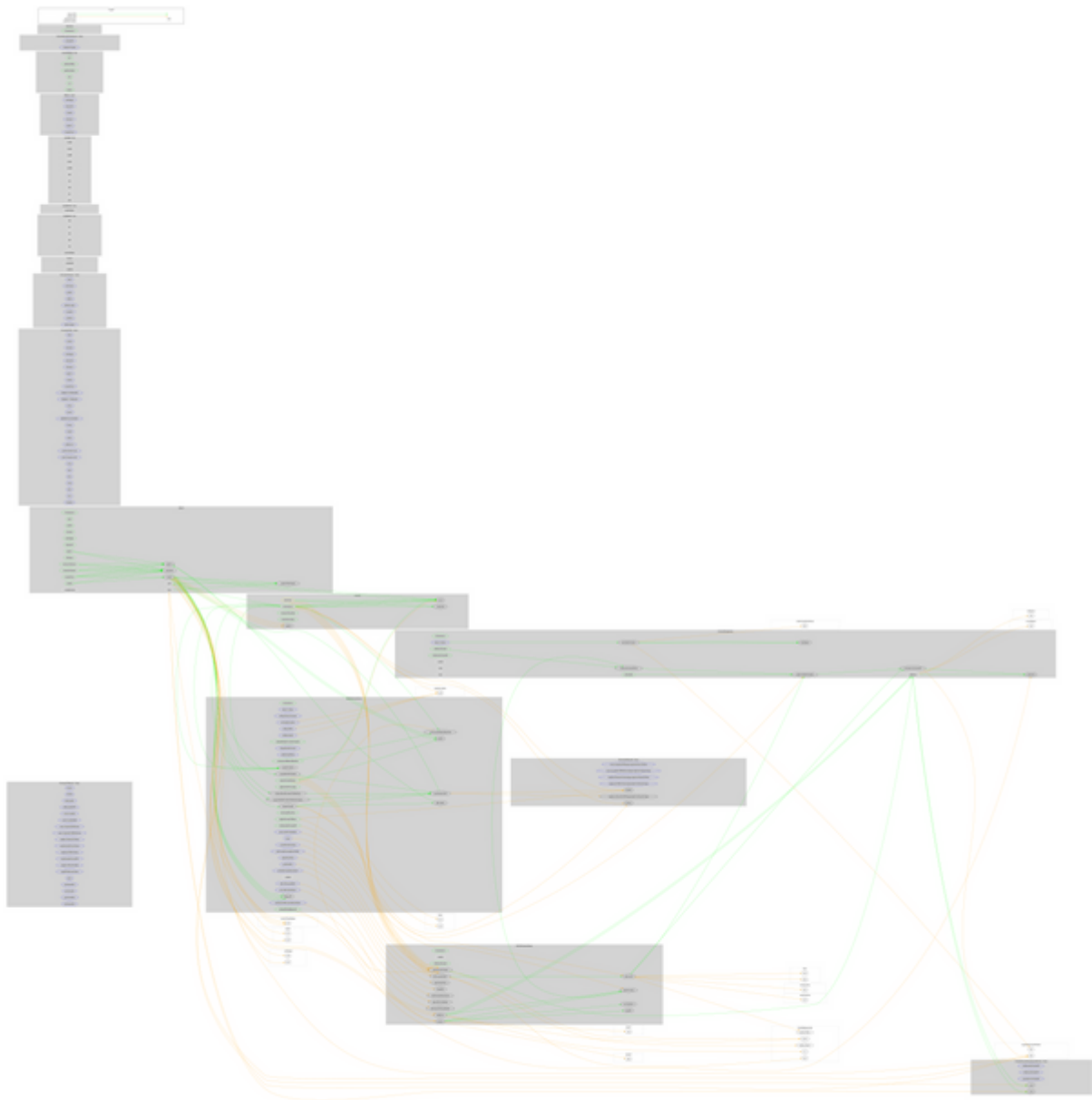
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
	_msgValue	Internal		
Auth	Implementation	Context		
	<Constructor>	Public	✓	-
	authorize	Public	✓	onlyOwner
	unauthorize	Public	✓	onlyOwner
	isOwner	Public		-
	isAuthorized	Public		-
IERC20Extended	Interface			
	name	External		-
	symbol	External		-
	decimals	External		-
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IUniswapV2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	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	✓	-
IRewardPoolDistributor	Interface			
	deposit	External	Payable	-

	distributeReward	External	✓	-
	withdrawReward	External	✓	-
RewardPoolDistributor	Implementation	IRewardPoolDistributor, Auth		
	<Constructor>	Public	✓	Auth
	unInitialized	External	✓	onlyToken
	setTokenAddress	External	✓	initializer onlyToken
	setDailyLimit	External	✓	authorized
	setTimeLimit	External	✓	authorized
	migratePool	External	✓	onlyOwner
	deposit	External	Payable	authorized onlyTokenAnd Owner
	distributeReward	External	✓	authorized
	withdrawReward	External	✓	-
	resetTimeLimit	Internal	✓	
	needResetTimeLimit	Internal		
Apocalypse	Implementation	IERC20Extended, Auth		
	<Constructor>	Public	Payable	Auth
	<Receive Ether>	External	Payable	-
	getCirculatingSupply	Public		-
	name	External		-
	symbol	External		-
	decimals	External		-
	totalSupply	External		-
	balanceOf	Public		-
	allowance	External		-
	approve	Public	✓	-
	approveMax	External	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-
	_transferFrom	Internal	✓	

	_basicTransfer	Internal	✓	
	shouldTakeFee	Internal		
	shouldSwapBack	Internal		
	shouldAutoBuyback	Internal		
	isOverLiquified	Public		-
	_initializeFees	Internal	✓	
	setFees	Public	✓	authorized
	_setFees	Internal	✓	
	setFeeReceivers	External	✓	authorized
	setIsFeeExempt	External	✓	authorized
	getTotalFee	Public		-
	getMultipliedFee	Public		-
	takeFee	Internal	✓	
	_initializeDistributor	Internal	✓	
	distributorInitialization	Public	✓	authorized
	setRewardPoolDistributor	Public	✓	authorized
	_initializeLiquidityBuyBack	Internal	✓	
	getLiquidityBacking	Public		-
	setTargetLiquidity	External	✓	authorized
	setSwapBackSettings	External	✓	authorized
	triggerSwapBack	External	✓	authorized
	swapBack	Internal	✓	swapping
	setAutoBuybackSettings	External	✓	authorized
	setBuybackMultiplierSettings	External	✓	authorized
	setBuyBacker	External	✓	authorized
	clearBuybackMultiplier	External	✓	authorized
	triggerAutoBuyback	Internal	✓	
	triggerZeusBuyback	External	✓	authorized
	buyTokens	Internal	✓	swapping

Contract Flow



Summary

The Smart Contract analysis reported no compiler error or critical issues. The contract Owner can access some admin functions that can not be used in a malicious way to disturb the users' transactions. There is also a limit of max 10% fees.

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.

CoinScope 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 CoinScope 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 CoinScope 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 CoinScope team disclaims any liability for the resulting losses.

About Coinscope

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.

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



The Coinscope.co team

<https://www.coinscope.co>