



Audit Report

All in

February 2022

Type	BEP20
Network	BSC
Address	0x05cd34a1d5df5010f5edb3412077b3c291ddb7a8
Audited by	© coinscope

Table of Contents

Table of Contents	1
Contract Review	3
Audit Updates	3
Contract Analysis	4
ST - Stop Transactions	5
Description	5
Recommendation	5
ELFM - Exceed Limit Fees Manipulation	6
Description	6
Recommendation	6
BC - Blacklisted Contracts	7
Description	7
Recommendation	7
Contract Diagnostics	8
L01 - Public Function could be Declared External	9
Description	9
Recommendation	9
L02 - State Variables could be Declared Constant	10
Description	10
Recommendation	10
L05 - Unused State Variable	11
Description	11
Recommendation	11
L04 - Conformance to Solidity Naming Conventions	12
Description	12
Recommendation	12

L07 - Missing Events Arithmetic	13
Description	13
Recommendation	13
Contract Functions	14
Contract Flow	18
Domain Info	19
Summary	20
Disclaimer	21
About Coinscope	22

Contract Review

Contract Name	Allin
Compiler Version	v0.8.9+commit.e5eed63a
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0x05cd34a1d5df5010f5edb3412077b3c291ddb7a8
Symbol	allin
Decimals	18
Total Supply	99,999,999,999
Source	contract.sol
Domain	allincryptogames.com

Audit Updates

Initial Audit	21st 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

ST - Stop Transactions

Criticality	medium
Location	contract.sol#L429

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 `tradingOpen` to false.

```
if(!authorizations[sender] && !authorizations[recipient]){  
    require(tradingOpen,"Trading not open yet");  
}
```

There is also a `_maxWalletToken` variable that can be set to any value. If the owner configures this variable to be a low number it can potentially fail transactions as well.

Recommendation

The contract could embody a check for not allowing setting the `_maxWalletToken` less than a reasonable amount. A suggested implementation could check that the maximum 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.

ELFM - Exceed Limit Fees Manipulation

Criticality	critical
Location	contract.sol#L636

Description

The contract owner has the authority to increase over the allowed limit of 25%. The owner may take advantage of it by calling the `setFees` function with high percentage values.

```
function setFees(uint256 _liquidityFee, uint256 _buybackFee, uint256
_reflectionFee, uint256 _marketingdevelopmentFee, uint256 _feeDenominator)
external authorized {
    liquidityFee = _liquidityFee;
    buybackFee = _buybackFee;
    reflectionFee = _reflectionFee;
    marketingdevelopmentFee = _marketingdevelopmentFee;
    totalFee =
    _liquidityFee.add(_buybackFee).add(_reflectionFee).add(_marketingdevelopmentFee
);
    feeDenominator = _feeDenominator;
}
```

Recommendation

The contract could embody a check for the maximum acceptable value.

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.

BC - Blacklisted Contracts

Criticality	minor
Location	contract.sol#L468

Description

The contract owner has the authority to stop contracts from buys for a specific period of time . The owner may take advantage of it by calling the `setIsTimelockExempt` function.

```
if (sender == pair &&
    buyCooldownEnabled &&
    !isTimelockExempt[recipient]) {
    require(cooldownTimer[recipient] < block.timestamp, "Please wait for
cooldown between buys");
    cooldownTimer[recipient] = block.timestamp + cooldownTimerInterval;
```

Recommendation

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
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L05	Unused State Variable
●	L04	Conformance to Solidity Naming Conventions
●	L07	Missing Events Arithmetic

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L77,79,85,673,695,700

Description

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

```
cooldownEnabled  
tradingStatus  
getUnpaidEarnings  
transferOwnership  
unauthorize  
authorize
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality

minor

Location

contract.sol#L303,305,304,306,312,159,172

Description

Constant state variables should be declared constant to save gas.

```
dividendsPerShareAccuracyFactor  
WBNB  
_totalSupply  
ZERO  
DEAD  
BUSD
```

Recommendation

Add the constant attribute to state variables that never change.

L05 - Unused State Variable

Criticality

minor

Location

contract.sol#L303

Description

There are segments that contain unused state variables.

BUSD

Recommendation

Remove unused state variables.

L04 - Conformance to Solidity Naming Conventions

Criticality

minor

Location

contract.sol#L99,197,150,158,159,588,636,645,650,655 and 29 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.

```
_allowances  
_balances  
_maxWalletToken  
_maxTxAmount  
_totalSupply  
_decimals  
_symbol  
_name  
ZERO  
...
```

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#L197,588,598,613,636,650,655

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.

```
targetLiquidity = _target
swapThreshold = _amount
liquidityFee = _liquidityFee
_maxTxAmount = amount
buybackMultiplierNumerator = numerator
autoBuybackCap = _cap
minPeriod = _minPeriod
```

Recommendation

Emit an event for critical parameter changes.

Contract Functions

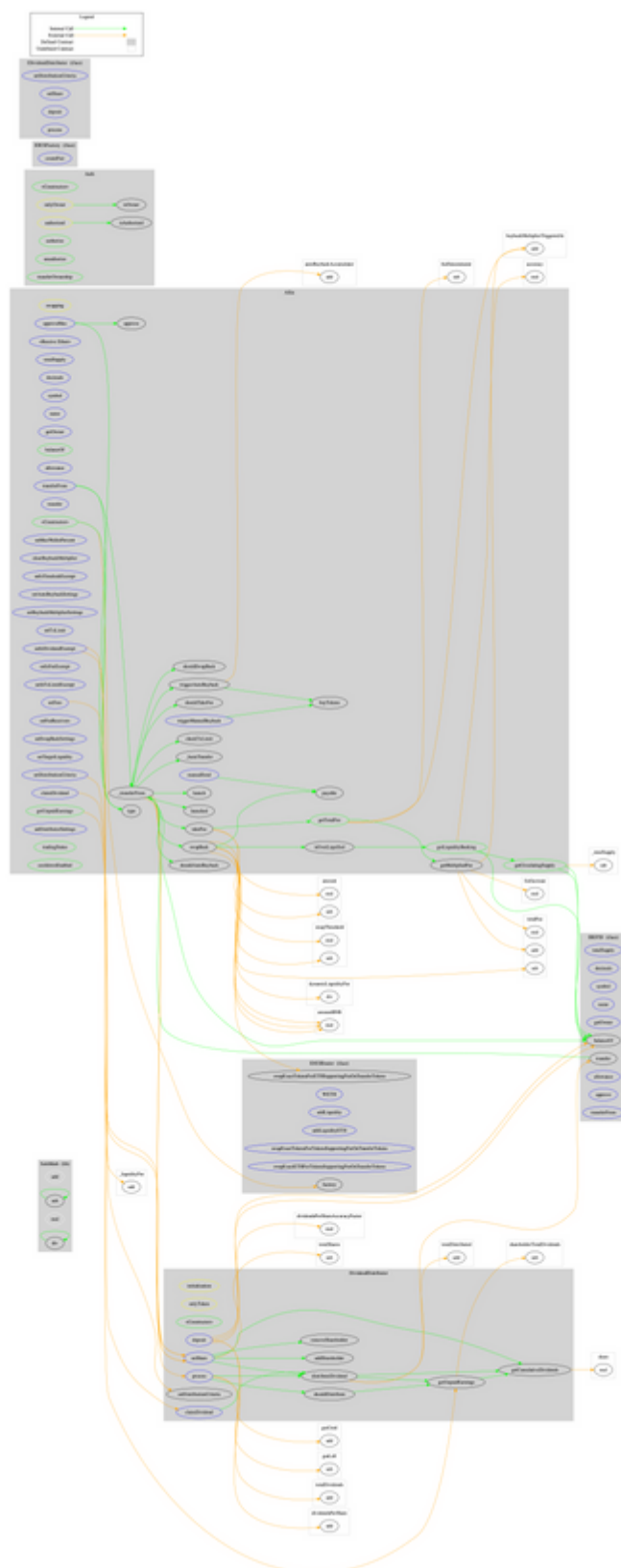
Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
SafeMath	Library			
	add	Internal		
	sub	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	div	Internal		
IBEP20	Interface			
	totalSupply	External		-
	decimals	External		-
	symbol	External		-
	name	External		-
	getOwner	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
Auth	Implementation			
	<Constructor>	Public	✓	-
	authorize	Public	✓	onlyOwner
	unauthorize	Public	✓	onlyOwner
	isOwner	Public		-
	isAuthorized	Public		-
	transferOwnership	Public	✓	onlyOwner

IDEXFactory	Interface			
	createPair	External	✓	-
IDEXRouter	Interface			
	factory	External		-
	WETH	External		-
	addLiquidity	External	✓	-
	addLiquidityETH	External	Payable	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
IDividendDistributor	Interface			
	setDistributionCriteria	External	✓	-
	setShare	External	✓	-
	deposit	External	Payable	-
	process	External	✓	-
DividendDistributor	Implementation	IDividendDistributor		
	<Constructor>	Public	✓	-
	setDistributionCriteria	External	✓	onlyToken
	setShare	External	✓	onlyToken
	deposit	External	Payable	onlyToken
	process	External	✓	onlyToken
	shouldDistribute	Internal		
	distributeDividend	Internal	✓	
	claimDividend	External	✓	onlyToken
	getUnpaidEarnings	Public		-
	getCumulativeDividends	Internal		
	addShareholder	Internal	✓	
	removeShareholder	Internal	✓	

Allin	Implementation	IBEP20, Auth		
	<Constructor>	Public	✓	Auth
	<Receive Ether>	External	Payable	-
	totalSupply	External		-
	decimals	External		-
	symbol	External		-
	name	External		-
	getOwner	External		-
	balanceOf	Public		-
	allowance	External		-
	approve	Public	✓	-
	approveMax	External	✓	-
	transfer	External	✓	-
	transferFrom	External	✓	-
	setMaxWalletPercent	External	✓	onlyOwner
	_transferFrom	Internal	✓	
	_basicTransfer	Internal	✓	
	checkTxLimit	Internal		
	shouldTakeFee	Internal		
	getTotalFee	Public		-
	getMultipliedFee	Public		-
	takeFee	Internal	✓	
	shouldSwapBack	Internal		
	swapBack	Internal	✓	swapping
	shouldAutoBuyback	Internal		
	triggerManualBuyback	External	✓	authorized
	clearBuybackMultiplier	External	✓	authorized
	triggerAutoBuyback	Internal	✓	
	buyTokens	Internal	✓	swapping
	setIsTimelockExempt	External	✓	authorized
	setAutoBuybackSettings	External	✓	authorized
	setBuybackMultiplierSettings	External	✓	authorized
	launched	Internal		
	launch	Internal	✓	

	setTxLimit	External	✓	authorized
	setIsDividendExempt	External	✓	authorized
	setIsFeeExempt	External	✓	authorized
	setIsTxLimitExempt	External	✓	authorized
	setFees	External	✓	authorized
	setFeeReceivers	External	✓	authorized
	setSwapBackSettings	External	✓	authorized
	setTargetLiquidity	External	✓	authorized
	manualSend	External	✓	authorized
	setDistributionCriteria	External	✓	authorized
	claimDividend	External	✓	-
	getUnpaidEarnings	Public		-
	setDistributorSettings	External	✓	authorized
	getCirculatingSupply	Public		-
	getLiquidityBacking	Public		-
	isOverLiquified	Public		-
	tradingStatus	Public	✓	onlyOwner
	cooldownEnabled	Public	✓	onlyOwner

Contract Flow



Domain Info

Domain Name	allincryptogames.com
Registry Domain ID	2662421445_DOMAIN_COM-VRSN
Creation Date	2021-12-18T12:35:45Z
Updated Date	2021-12-18T13:36:33Z
Registry Expiry Date	
Registrar WHOIS Server	whois.registrar.eu
Registrar URL	http://www.registrar.eu
Registrar	Hosting Concepts B.V. d/b/a Registrar.eu
Registrar IANA ID	1647

The domain has been created 2 months before the creation of the audit.

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

Summary

All in token aims to be a gambling platform that lets users earn crypto tokens. The Project has a friendly and growing community. There are some functions that can be abused by the owner, like manipulating fees up to 100%. The owner can also pause transactions and timeban users from buying. Finally there is a maximum buy amount that a user can have. 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.

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>