COINSCOPE

Audit Report

Tick a Lock

January 2022

Type	BEP20
Network	BSC
Address	0x1252C3d8770d13A38065848a7476964142D848Fe
Audited by	© coinscope

Table of Contents

Contract Review

Contract Name	TickALock
Compiler Version	v0.8.11+commit.d7f03943
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0x1252C3d8770d13A38065848a7476964142D848Fe
Symbol	TIALO
Decimals	9
Total Supply	24,000,000,000
Source	contract.sol
Domain	tickalock.app

Audit Updates

Initial Audit	24th January 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	critical
Location	contract.sol#L944

Description

The contract owner has the authority to stop all the sales excluding the owner. The owner may take advantage of it by setting to the `maxBurnTax` a value that is lower than the `minBurnTax` . This will cause the following expression to produce a negative number.

```
burnTax = ((maxBurnTax - minBurnTax) * relativePercentageValue) /
10000 + minBurnTax;
```

Thus, the `_sellTokens` function that is calling the `calculateBurnTax` will fail.

```
function calculateBurnTax(address sender) public view returns (uint256) {
    uint256 burnTax = 0;
    //get senders balance
    uint256 senderBalance = _balances[sender];
    //max burn tax defaults to %7.24
    uint256 maxBurnTax = _maxBurnTax;
    //min burn tax defaults to %0.24
    uint256 minBurnTax = _minBurnTax;
    uint256 percentageOfHoldings = (senderBalance * 10000) / _totalSupply;
    //given that the percentageOfHoldings can only ever be max 1% of the
circulating supply, make this a percentage where the max is 100%
    uint256 relativePercentageValue = percentageOfHoldings * 100;
    //given the range of min minBurnTax and max maxBurnTax, find the relative
percentage of the percentage of holdings.  This is the burn tax.
    burnTax = ((maxBurnTax - minBurnTax) * relativePercentageValue) / 10000 +
minBurnTax;

    return burnTax;
}
```

Recommendation

The contract could embody a check for not allowing setting the `maxBurnTax` less than the `minBurnTax`. 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.

Contract Diagnostics

● Critical ● Medium ● Minor

Severity	Code	Description
●	MAL	Misused Algorithmic Logic
●	CR	Code Repetition
●	CO	Code Optimization
●	L01	Public Function could be Declared External
●	L02	State Variables could be Declared Constant
●	L05	Unused State Variable
●	L04	Conformance to Solidity Naming Conventions
●	L09	Dead Code Elimination
●	L07	Missing Events Arithmetic
●	L08	Tautology or Contradiction

MAL - Misused Algorithmic Logic

Criticality	minor
Location	contract.sol#L1363

Description

The algorithmic flow does not follow the required business logic.

```
string memory lowercaseLetter = _toLower(letter);
//bool if letter is in the alphabet
bool isInAlphabet = false;
//iterate through letters and see if lowercaseLetter is in the array
for(uint256 i = 0; i < _alphabet.length; i++) {
    if(compareStrings(lowercaseLetter, _alphabet[i])) {
        isInAlphabet = true;
    }
}
require(
    isInAlphabet,
    "Letter does not exist in the alphabet"
);
```

The algorithm is aiming to determine if the letter contains a letter that is not alphanumeric. There are some cases that do not produce the expected result. For instance:

Input	Result	Expected Result
abc	true	true
x_	false	false
a_b	**true**	**false**

Recommendation

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

CR - Code Repetition

Criticality	minor
Location	contract.sol#L1473

Description

There are code segments that are repetitive in the contract. Those segments increase the code size of the contract unnecessarily.

```
_puzzleGuesses[_currentPuzzleWeek][guessHash] = Guess({
        guesser: msg.sender,
        guess: _guess,
        timestamp: block.timestamp,
        hasBeenSubmitted: true
    });
_puzzleWeekFullGuessList[_currentPuzzleWeek].push(
    Guess({
        guesser: msg.sender,
        guess: _guess,
        timestamp: block.timestamp,
        hasBeenSubmitted: true
    })
);
```

The Guess instance is created twice with exactly the same state.

Recommendation

The guess structure could be created once and assign the same instance to both arrays.

CO - Code Optimization

Criticality	minor
Location	contract.sol#L1378

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.

There are expressions that are repetitively called in the same method and always yield the same result. For instance, the `getDeterministicCoordinateHash()` function that is called by the `submitCrosswordLetter()` function.

```
if(_coordinatesForWeek[_currentPuzzleWeek][getDeterministicCoordinateHash(xCoord
inate, yCoordinate)].x != 0 &&
_coordinatesForWeek[_currentPuzzleWeek][getDeterministicCoordinateHash(xCoordina
te, yCoordinate)].y != 0) {
...
if(_crossWordLetterGuesses[_currentPuzzleWeek][getDeterministicCoordinateHash(xC
oordinate, yCoordinate)][lowercaseLetter].isLetterGuessed) {
...
require(_guessCountOfCoordinate[_currentPuzzleWeek][getDeterministicCoordinateHa
sh(xCoordinate, yCoordinate)][msg.sender] < _crosswordGuessesPerCoordinate, "You
have exceeded your guesses for this coordinate.");
...
_crossWordLetterGuesses[_currentPuzzleWeek][getDeterministicCoordinateHash(xCoor
dinate, yCoordinate)][lowercaseLetter] = LetterGuessed(
...
_guessCountOfCoordinate[_currentPuzzleWeek][getDeterministicCoordinateHash(xCoor
dinate, yCoordinate)][msg.sender]++;
```

Recommendation

The repetitive expressions could be calculated once and reuse the same result for the next expressions.

L01 - Public Function could be Declared External

Criticality	minor
Location	contract.sol#L1505,L1454,L1450 and 24 more

Description

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

```
setPuzzleSolved
submitGuess
setMinGuessLength
...
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contract.sol#L711,L751,L723 and 6 more

Description

Constant state variables should be declared constant to save gas.

```
_totalSupply
_pancakeRouterAddress
_numTokensSellToAddToLiquidity
...
```

Recommendation

Add the constant attribute to state variables that never change.

L05 - Unused State Variable

Criticality	minor
Location	contract.sol#L720

Description

There are segments that contains unused state variable.

```
_isWithdrawing
```

Recommendation

Remove unused state variables.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contract.sol#L1321,L1319,L1317 and 44 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.

```
_guessCountOfCoordinate
_crossWordLetterGuesses
_coordinatesForWeek
...
```

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#L676,L568,L622 and 28 more

Description

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

```
remove
length
contains
...
```

Recommendation

Remove unused functions.

L07 - Missing Events Arithmetic

Criticality	minor
Location	contract.sol#L1450,L1446,L1323 and 1 more

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.

```
_minGuessLength = minGuessLength
_minHoldingsForGuess = minHoldingsForGuess * 10 ** 9
_guessCount = guessCount
...
```

Recommendation

Emit an event for critical parameter changes.

L08 - Tautology or Contradiction

Criticality	minor
Location	contract.sol#L961

Description

Detects expressions that are tautologies or contradictions. For instance, an uint variable will always be greater than or equal to zero.

```
require(bool,string)(_marketingTax >= 0 && _marketingTax <= 400,Marketing tax
must be between 0 and 400)
```

Recommendation

Fix the incorrect comparison by changing the value type or the comparison.

Contract Functions

Contract	Type	Bases		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
IBEP20	Interface			
	totalSupply	External		-
	decimals	External		-
	symbol	External		-
	name	External		-
	getOwner	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IPancakeERC20	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	✓	-
Ownable	Implementation			

	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	_verifyCallResult	Private		
IPancakeFactory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
IPancakeRouter01	Interface			
	addLiquidity	External	✓	-
	addLiquidityETH	External	Payable	-
	removeLiquidity	External	✓	-
	removeLiquidityETH	External	✓	-
	removeLiquidityWithPermit	External	✓	-

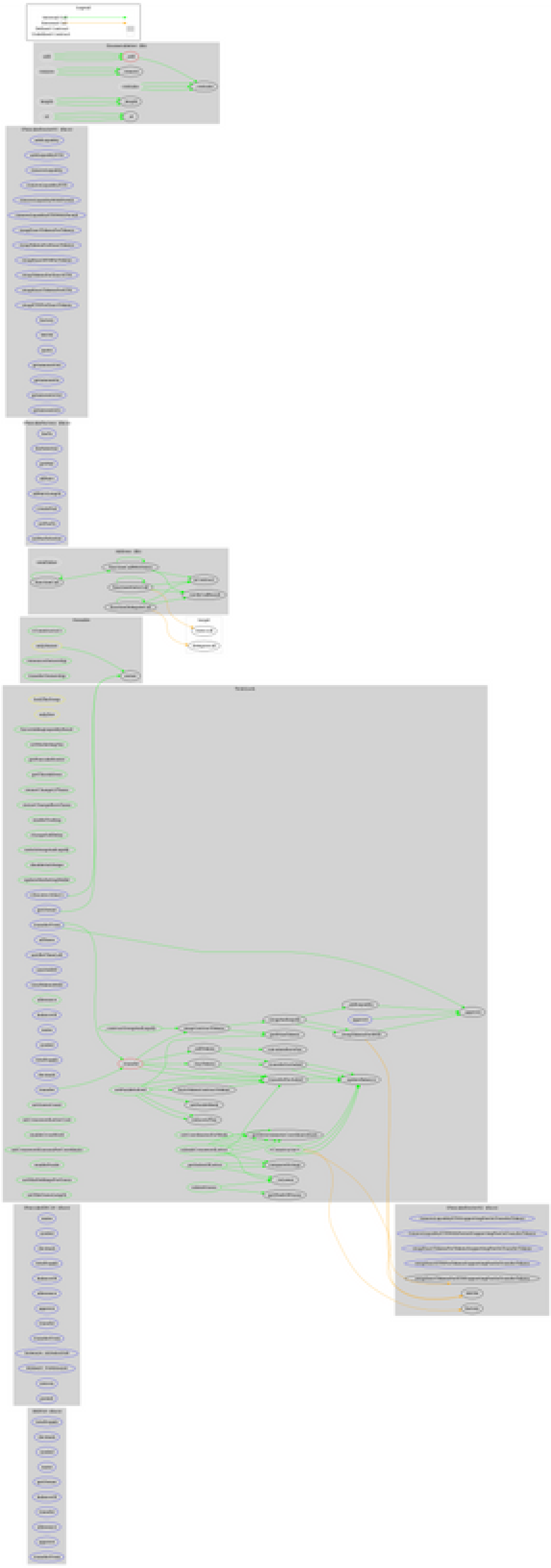
	removeLiquidityETHWithPermit	External	✓	-
	swapExactTokensForTokens	External	✓	-
	swapTokensForExactTokens	External	✓	-
	swapExactETHForTokens	External	Payable	-
	swapTokensForExactETH	External	✓	-
	swapExactTokensForETH	External	✓	-
	swapETHForExactTokens	External	Payable	-
	factory	External		-
	WETH	External		-
	quote	External		-
	getamountOut	External		-
	getamountIn	External		-
	getamountsOut	External		-
	getamountsIn	External		-
IPancakeRouter02	Interface	IPancakeRouter01		
	removeLiquidityETHSupportingFeeOnTransferTokens	External	✓	-
	removeLiquidityETHWithPermitSupportingFeeOnTransferTokens	External	✓	-
	swapExactTokensForTokensSupportingFeeOnTransferTokens	External	✓	-
	swapExactETHForTokensSupportingFeeOnTransferTokens	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-
EnumerableSet	Library			
	_add	Private	✓	
	_remove	Private	✓	
	_contains	Private		
	_length	Private		
	_at	Private		
	add	Internal	✓	
	remove	Internal	✓	
	contains	Internal		

	length	Internal		
	at	Internal		
	add	Internal	✓	
	remove	Internal	✓	
	contains	Internal		
	length	Internal		
	at	Internal		
	add	Internal	✓	
	remove	Internal	✓	
	contains	Internal		
	length	Internal		
	at	Internal		
TickALock	Implementation	Ownable, IBEP20		
	<Constructor>	Public	✓	-
	forceAddingLiquidityReset	Public	✓	onlyDev
	_transfer	Private	✓	
	_buyTokens	Private	✓	
	_sellTokens	Private	✓	
	calculateBurnTax	Public		-
	setMarketingTax	Public	✓	onlyOwner
	_transferIncluded	Private	✓	
	_transferExcluded	Private	✓	
	_updateBalance	Private	✓	
	_contractSwapAndLiquify	Public	✓	onlyDev
	swapContractTokens	Private	✓	lockTheSwap
	getPrizeTokens	Public		-
	swapAndLiquify	Private	✓	
	_addLiquidity	Private	✓	
	_swapTokensForBNB	Private	✓	
	getPancakeRouter	Public		-
	getThisAddress	Public		-
	reduceLPTax	Private	✓	
	ownerChangeLPTaxes	Public	✓	onlyOwner
	ownerChangeBurnTaxes	Public	✓	onlyOwner

	enableTrading	Public	✓	onlyOwner
	changeSellDelay	Public	✓	onlyOwner
	switchSwapAndLiquify	Public	✓	onlyOwner
	disableAntiSnipe	Public	✓	onlyOwner
	updateMarketingWallet	Public	✓	onlyOwner
	_approve	Private	✓	
	transfer	External	✓	-
	transferFrom	External	✓	-
	approve	External	✓	-
	<Receive Ether>	External	Payable	-
	allTaxes	External		-
	antiBotTimeLeft	External		-
	nextSellOf	External		-
	totalTokensHeld	External		-
	allowance	Public		-
	balanceOf	External		-
	name	External		-
	symbol	External		-
	totalSupply	External		-
	decimals	External		-
	getOwner	External		-
	getDeterministicCoordinateHash	Public		-
	setGuessCount	Public	✓	onlyDev
	setCoordinatesForWeek	Public	✓	onlyDev
	setCrosswordLetterCost	Public	✓	onlyDev
	enableCrossWord	Public	✓	onlyDev
	setCrosswordGuessesPerCoordinate	Public	✓	onlyDev
	enablePuzzle	Public	✓	onlyDev
	submitCrosswordLetter	Public	Payable	-
	_getIndexOfLetter	Public		-
	compareStrings	Private		
	_toLower	Internal		
	setMinHoldingsForGuess	Public	✓	onlyDev
	setMinGuessLength	Public	✓	onlyDev
	submitGuess	Public	✓	-

	getHashOfGuess	Public		-
	setPuzzleWeek	Private	✓	
	setPuzzleSolved	Public	✓	onlyDev
	burnTokenContractTokens	Private	✓	

Contract Flow



Domain Info

Domain Name	tickalock.app
Registry Domain ID	483BFE666-APP
Creation Date	2021-12-08T23:12:05Z
Updated Date	2021-12-22T01:52:38Z
Registry Expiry Date	2022-12-08T23:12:05Z
Registrar WHOIS Server	whois.google.com
Registrar URL	domains.google
Registrar	Google LLC.
Registrar IANA ID	895

The domain has been created about 2 months before the creation of the audit. It will expire in 11 months.

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

Summary

Tick a Lock combines tokenimics with gamification in the same contract. The Smart Contract analysis reported one issue. The contract owner can stop the sales by misusing the state of two variables. Additionally, the contract contains a check that it can be enabled by the contract owner and prevent users from selling in a timespan less than one hour.

The smart contract contains a game where users submit solutions for a puzzle. The users have to pay a fee in order to submit the solution. This fee is accumulated to the smart contract. Proportions of the accumulated fee is moved to the marketing, liquidity and crossword pot vaults. The crossword pot is the award that is shared to the winners.

The contract owner is responsible for setting the correct answer and the list of winners. The smart contract does not provide any guarantee that all the correct solutions will be awarded.

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