Cyberscope

Audit Report

Elons Roadmap

June 2023

Network	BSC
Address	0xb83E7B111b2986b423361a351c4936061Fda5Ab9
Audited by	© cyberscope

Analysis

● Critical ● Medium ● Minor / Informative ● Pass

Severity	Code	Description	Status
●	ST	Stops Transactions	Passed
●	OTUT	Transfers User's Tokens	Passed
●	ELFM	Exceeds Fees Limit	Passed
●	MT	Mints Tokens	Passed
●	BT	Burns Tokens	Passed
●	BC	Blacklists Addresses	Passed

Diagnostics

🔴 Critical 🟠 Medium ● Minor / Informative

Severity	Code	Description	Status
●	RPF	Redundant Private Function	Unresolved
●	RSW	Redundant Storage Writes	Unresolved
●	PVC	Price Volatility Concern	Unresolved
●	RSD	Redundant Struct Declaration	Unresolved
●	RVD	Redundant Variable Declaration	Unresolved
●	MMN	Misleading Modifier Naming	Unresolved

Table of Contents

Review

Contract Name	ElonsRoadmap
Compiler Version	v0.8.0+commit.c7dfd78e
Optimization	200 runs
Explorer	https://bscscan.com/address/0xb83e7b111b2986b423361a351c4936061fda5ab9
Address	0xb83e7b111b2986b423361a351c4936061fda5ab9
Network	BSC
Symbol	ELMAP
Decimals	9
Total Supply	1.000.000.000

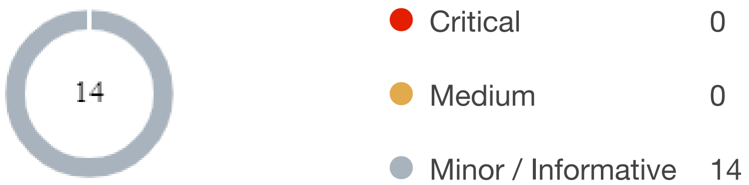
Audit Updates

Initial Audit	28 May 2023 https://github.com/cyberscope-io/audits/blob/main/elmap/v1/audit.pdf
Corrected Phase 2	03 Jun 2023 https://github.com/cyberscope-io/audits/blob/main/elmap/v2/audit.pdf
Corrected Phase 3	10 Jun 2023

Source Files

Filename	SHA256
ElonsRoadmap.sol	6eab9380532ef6ef2de92cd0fb3ac521990062f5208bdeec99169293480a21b8

Findings Breakdown



Severity	Unresolved	Acknowledged	Resolved	Other
● Critical	0	0	0	0
● Medium	0	0	0	0
● Minor / Informative	14	0	0	0

RSW - Redundant Storage Writes

Criticality	Minor / Informative
Location	ElonsRoadmap.sol#L470,555,559,562,782,785,788
Status	Unresolved

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.

The contract modifies the state of some variables without checking if the current state of these variables is the same as the one given as an argument. As a result, the contract performs redundant storage writes.

```
function setMoveBnbToWallets(bool state) external onlyOwner {
    moveBnbToWallets = state;
}

function excludeFromFee(address account) public onlyOwner {
    excludedFromFees[account] = true;
}

function includeInFee(address account) public onlyOwner {
    excludedFromFees[account] = false;
}
...
```

Recommendation

The team is advised to take these segments into consideration and rewrite them so the runtime will be more performant. That way it will improve the efficiency and performance of the source code and reduce the cost of executing it.

PVC - Price Volatility Concern

Criticality	Minor / Informative
Location	ElonsRoadmap.sol#L777
Status	Unresolved

Description

The contract accumulates tokens from the taxes to swap them for ETH. The variable `minimumTokensBeforeSwap` sets a threshold where the contract will trigger the swap functionality. If the variable is set to a big number, then the contract will swap a huge amount of tokens for ETH.

It is important to note that the price of the token representing it, can be highly volatile. This means that the value of a price volatility swap involving Ether could fluctuate significantly at the triggered point, potentially leading to significant price volatility for the parties involved.

```
function setSwapAndLiquify(bool _state, uint _minimumTokensBeforeSwap)
external onlyOwner {
    swapAndLiquifyEnabled = _state;
    minimumTokensBeforeSwap = _minimumTokensBeforeSwap;
}
```

Recommendation

The contract could ensure that it will not sell more than a reasonable amount of tokens in a single transaction. A suggested implementation could check that the maximum amount should be less than a fixed percentage of the total supply. Hence, the contract will guarantee that it cannot accumulate a huge amount of tokens in order to sell them.

RSD - Redundant Struct Declaration

Criticality	Minor / Informative
Location	ElonsRoadmap.sol#L417
Status	Unresolved

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.

The contract declares the `userData` struct to keep track of each user's last buy. Since the struct only contains one property, it could be omitted. As a result the struct is redundant.

```
struct userData {
    uint lastBuyTime;
}
mapping (address => userData) public userLastTradeData;
```

Recommendation

The team is advised to take these segments into consideration and rewrite them so the runtime will be more performant. That way it will improve the efficiency and performance of the source code and reduce the cost of executing it.

The contract could modify the `userLastTradeData` mapping to return a uint256 integer for each address instead of a struct.

```
mapping (address => uint256) public userLastTradeData;
```

RVD - Redundant Variable Declaration

Criticality	Minor / Informative
Location	ElonsRoadmap.sol#L378,385,401
Status	Unresolved

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.

The contract declares some variables that are not used in a meaningful way by the contract. As a result, these variables are redundant.

```
bool public TakeBnbForFees = true;
uint public maxSellTxAmount;
bool public marketActive = true;
```

Recommendation

The team is advised to take these segments into consideration and rewrite them so the runtime will be more performant. That way it will improve the efficiency and performance of the source code and reduce the cost of executing it.

MMN - Misleading Modifier Naming

Criticality	Minor / Informative
Location	ElonsRoadmap.sol#L773
Status	Unresolved

Description

Modifiers can have misleading names if their names do not accurately reflect the functionality they contain or the purpose they serve. The contract uses some modifier names that are too generic or do not clearly convey the underneath functionality. Misleading modifier names can lead to confusion, making the code more difficult to read and understand. modifiers can have misleading names if their names do not accurately reflect the functionality they contain or the purpose they serve. The contract uses some modifier names that are too generic or do not clearly convey the underneath functionality. Misleading modifier names can lead to confusion, making the code more difficult to read and understand.

The `FastTx` modifier is executed when the contract is swapping tokens. Hence, its name does not reflect its functionality.

```
modifier FastTx() {
    isInternalTransaction = true;
    _;
    isInternalTransaction = false;
}
```

Recommendation

It's always a good practice for the contract to contain modifier names that are specific and descriptive. The team is advised to keep in mind the readability of the code.

RPF - Redundant Private Function

Criticality	Minor / Informative
Location	ElonsRoadmap.sol#L566
Status	Unresolved

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.`

The contract implements the private `setFees` function. But it is not utilized in the contract's implementation.

```
function setFees() private {
    buyFee = buyReflectionFee + buyDevelopFee + buyLiqFee + buyMarketingFee;
    sellFee = sellReflectionFee + sellDevelopFee + sellLiqFee +
sellMarketingFee;
}
```

Recommendation

The team is advised to take these segments into consideration and rewrite them so the runtime will be more performant. That way it will improve the efficiency and performance of the source code and reduce the cost of executing it. It is recommended to remove redundant functions.

RSML - Redundant SafeMath Library

Criticality	Minor / Informative
Location	ElonsRoadmap.sol
Status	Unresolved

Description

SafeMath is a popular Solidity library that provides a set of functions for performing common arithmetic operations in a way that is resistant to integer overflows and underflows.

Starting with Solidity versions that are greater than or equal to 0.8.0, the arithmetic operations revert to underflow and overflow. As a result, the native functionality of the Solidity operations replaces the SafeMath library. Hence, the usage of the SafeMath library adds complexity, overhead and increases gas consumption unnecessarily.

```
library SafeMath {...}
```

Recommendation

The team is advised to remove the SafeMath library. Since the version of the contract is greater than `0.8.0` then the pure Solidity arithmetic operations produce the same result.

If the previous functionality is required, then the contract could exploit the `unchecked { ... }` statement.

Read more about the breaking change on https://docs.soliditylang.org/en/v0.8.16/08

Functions Analysis

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	Private		
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		
Ownable	Implementation	Context		
		Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	getTime	Public		-

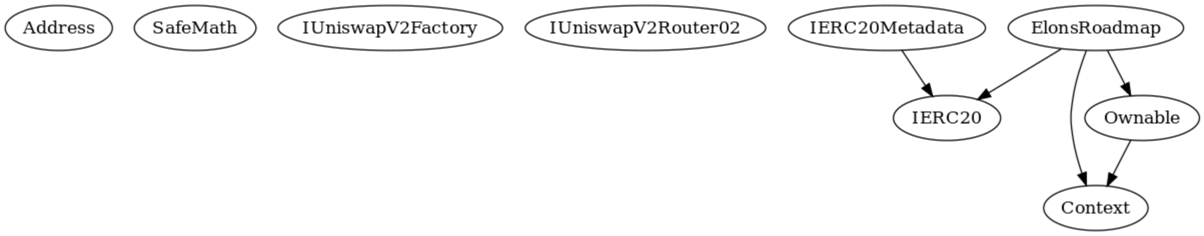
ElonsRoadmap	Implementation	Context, IERC20, Ownable		
		Public	✓	-
		External	Payable	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	setMoveBnbToWallets	External	✓	onlyOwner
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	isExcludedFromReward	Public		-
	totalFees	Public		-
	reflectionFromToken	Public		-
	tokenFromReflection	Public		-
	excludeFromReward	Public	✓	onlyOwner

	includeInReward	External	✓	onlyOwner
	excludeFromFee	Public	✓	onlyOwner
	includeInFee	Public	✓	onlyOwner
	setSwap	External	✓	onlyOwner
	setFees	Private	✓	
	setReflectionFee	External	✓	onlyOwner
	setDevelopFee	External	✓	onlyOwner
	setLiquidityFee	External	✓	onlyOwner
	setMarketingFee	External	✓	onlyOwner
	setMaxTxPercent	External	✓	onlyOwner
	_reflectFee	Private	✓	
	_getValues	Private		
	_getTValues	Private		
	_getRValues	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	_takeLiquidity	Private	✓	
	_takeDevelop	Private	✓	
	_takeMarketing	Private	✓	
	calculateReflectionFee	Private		
	calculateDevelopFee	Private		
	calculateLiquidityFee	Private		

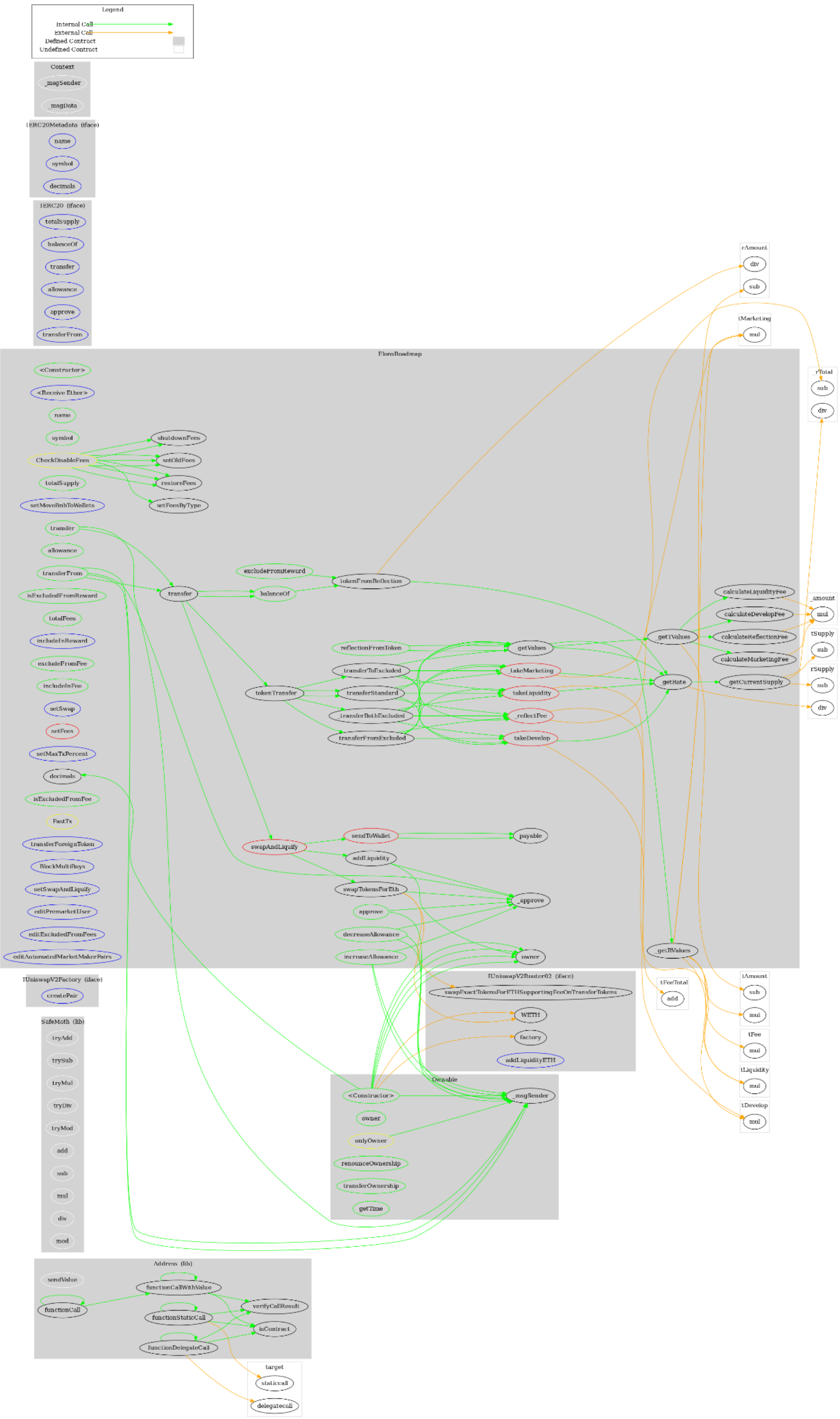
	calculateMarketingFee	Private		
	setOldFees	Private	✓	
	shutdownFees	Private	✓	
	setFeesByType	Private	✓	
	restoreFees	Private	✓	
	isExcludedFromFee	Public		-
	_approve	Private	✓	
	sendToWallet	Private	✓	
	swapAndLiquify	Private	✓	FastTx
	transferForeignToken	External	✓	onlyOwner
	BlockMultiBuys	External	✓	onlyOwner
	setSwapAndLiquify	External	✓	onlyOwner
	editPremarketUser	External	✓	onlyOwner
	editExcludedFromFees	External	✓	onlyOwner
	editAutomatedMarketMakerPairs	External	✓	onlyOwner
	swapTokensForEth	Private	✓	
	_transfer	Private	✓	
	addLiquidity	Private	✓	
	_tokenTransfer	Private	✓	CheckDisableFees
	_transferStandard	Private	✓	
	_transferToExcluded	Private	✓	

	_transferFromExcluded	Private	✓	
	_transferBothExcluded	Private	✓	

Inheritance Graph



Flow Graph


Legend
Internal Call
External Call
Defined Contract
Undefined Contract
Context
_msgSender
_msgData
IERC20Metadata (iface)
name
symbol
decimals
IERC20 (iface)
totalSupply
balanceOf
transfer
allowance
approve
transferFrom
ElonsRoadmap
<Constructor>
<Receive Ether>
name
symbol
CheckDisableFees
totalSupply
setMoveBnbToWallets
transfer
allowance
transferFrom
isExcludedFromReward
totalFees
includeInReward
excludeFromFee
includeInFee
setSwap
setFees
setMaxTxPercent
decimals
isExcludedFromFee
FastTx
transferForeignToken
BlackMultiBuys
setSwapAndLiquify
editPremarketUser
editExcludedFromFees
editAutomatedMarketMakerPairs
shutdownFees
setOldFees
restoreFees
setFeesByType
_transfer
excludeFromReward
balanceOf
tokenFromReflection
reflectionFromToken
_transferToExcluded
_tokenTransfer
_transferStandard
_transferBothExcluded
_transferFromExcluded
_getValues
_takeMarketing
_takeLiquidity
_reflectFee
_takeDevelop
_getTValues
_getRate
calculateLiquidityFee
calculateDevelopFee
calculateReflectionFee
calculateMarketingFee
_getCurrentSupply
swapAndLiquify
sendToWallet
addLiquidity
swapTokensForEth
approve
decreaseAllowance
increaseAllowance
payable
_approve
owner
_getRValues
rAmount
div
sub
tMarketing
mul
_rTotal
sub
div
_amount
mul
tSupply
sub
rSupply
sub
div
IUniswapV2Factory (iface)
createPair
IUniswapV2Router02 (iface)
swapExactTokensForETHSupportingFeeOnTransferTokens
WETH
factory
addLiquidityETH
tFeeTotal
add
tAmount
sub
mul
tFee
mul
tLiquidity
mul
tDevelop
mul
SafeMath (lib)
tryAdd
trySub
tryMul
tryDiv
tryMod
add
sub
mul
div
mod
Ownable
<Constructor>
_msgSender
owner
onlyOwner
renounceOwnership
transferOwnership
getTime
Address (lib)
sendValue
functionCall
functionCallWithValue
functionStaticCall
functionDelegateCall
verifyCallResult
isContract
target
staticcall
delegatecall

Summary

Elons Roadmap contract implements a token mechanism. This audit investigates security issues, business logic concerns, and potential improvements. Elons Roadmap is an interesting project that has a friendly and growing community. The Smart Contract analysis reported no compiler errors 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 fixed buy limit of a 4% fee and a fixed sell fee of 10%.

Disclaimer

The information provided in this report does not constitute investment, financial or trading advice and you should not treat any of the document's content as such. This report may not be transmitted, disclosed, referred to or relied upon by any person for any purposes nor may copies be delivered to any other person other than the Company without Cyberscope's prior written consent. This report is not nor should be considered an "endorsement" or "disapproval" of any particular project or team. This report is not nor should be regarded as an indication of the economics or value of any "product" or "asset" created by any team or project that contracts Cyberscope to perform a security assessment. This document does not provide any warranty or guarantee regarding the absolute bug-free nature of the technology analyzed, nor do they provide any indication of the technologies proprietors' business, business model or legal compliance. This report should not be used in any way to make decisions around investment or involvement with any particular project. This report represents an extensive assessment process intending to help our customers increase the quality of their code while reducing the high level of risk presented by cryptographic tokens and blockchain technology.

Blockchain technology and cryptographic assets present a high level of ongoing risk Cyberscope's position is that each company and individual are responsible for their own due diligence and continuous security Cyberscope's goal is to help reduce the attack vectors and the high level of variance associated with utilizing new and consistently changing technologies and in no way claims any guarantee of security or functionality of the technology we agree to analyze. The assessment services provided by Cyberscope are subject to dependencies and are under continuing development. You agree that your access and/or use including but not limited to any services reports and materials will be at your sole risk on an as-is where-is and as-available basis Cryptographic tokens are emergent technologies and carry with them high levels of technical risk and uncertainty. The assessment reports could include false positives false negatives and other unpredictable results. The services may access and depend upon multiple layers of third parties.

About Cyberscope

Cyberscope is a blockchain cybersecurity company that was founded with the vision to make web3.0 a safer place for investors and developers. Since its launch, it has worked with thousands of projects and is estimated to have secured tens of millions of investors' funds.

Cyberscope is one of the leading smart contract audit firms in the crypto space and has built a high-profile network of clients and partners.

The Cyberscope team

https://www.cyberscope.io