



Cyberscope

Audit Report

MidGard

March 2022

Type BEP20

Network BSC

Address 0x2260d68bf89894A0622eC31dE4D92105cc78A7d9

Audited by © cyberscope

Table of Contents

Table of Contents	1
Contract Review	3
Audit Updates	3
Contract Analysis	4
ULTW - Unlimited Liquidity to Team Wallet	5
Description	5
Recommendation	6
Contract Diagnostics	7
L01 - Public Function could be Declared External	8
Description	8
Recommendation	8
L02 - State Variables could be Declared Constant	9
Description	9
Recommendation	9
L04 - Conformance to Solidity Naming Conventions	10
Description	10
Recommendation	10
L09 - Dead Code Elimination	11
Description	11
Recommendation	11
L07 - Missing Events Arithmetic	12
Description	12
Recommendation	12
Contract Functions	13
Contract Flow	18
Domain Info	19

Summary	20
Disclaimer	21
About Cyberscope	22

Contract Review

Contract Name	MidGard
Compiler Version	v0.8.12+commit.f00d7308
Optimization	200 runs
Licence	Unlicense
Explorer	https://bscscan.com/token/0x2260d68bf89894A0622eC31dE4D92105cc78A7d9
Symbol	EPS
Decimals	9
Total Supply	100,000,000
Source	contract.sol
Domain	mid-gard-online.com

Audit Updates

Initial Audit	6th March 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

ULTW - Unlimited Liquidity to Team Wallet

Criticality	medium
Location	contract.sol#L851

Description

The contract owner has the authority to transfer a lot of funds to the team wallet. These funds have been accumulated from fees collected from the contract. Since the fees are fixed and can be changed, all the accumulated amount is converted to BNB and transferred to the team's wallet.

As a result the `swapAndLiquify` feature is useless since it will never be triggered.

```
feeRatesStruct public buyRates = feeRatesStruct(
    {marketing: 50,
     liquidity: 0
});

feeRatesStruct public sellRates = feeRatesStruct(
    {marketing: 70,
     liquidity: 0
});
```

```
if (contractBalance.lp_balance >= numTokensToSwap && !inSwapAndLiquify && from !=
pancakeswapV2Pair && swapAndLiquifyEnabled) {
    swapAndLiquify(numTokensToSwap);
}

if (contractBalance.marketing_balance >= numTokensToSwap && !inSwapAndLiquify &&
from != pancakeswapV2Pair && swapAndLiquifyEnabled) {
    swapAndSendToMarketing(numTokensToSwap);
}
```

Recommendation

The swapAndLiquify feature could be removed from the contract since it cannot be executed. The contract should embody a check for not allowing high liquidations to the team's wallet.

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
●	L04	Conformance to Solidity Naming Conventions
●	L09	Dead Code Elimination
●	L07	Missing Events Arithmetic

L01 - Public Function could be Declared External

Criticality

minor

Location

contract.sol#L277,282,640,644,648,652,656,665,674,679 and 10 more

Description

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

```
withdraw  
isExcludedFromFee  
includeInFee  
excludeFromFee  
excludeFromAll  
excludeFromReward  
reflectionFromToken  
isExcludedFromReward  
decreaseAllowance  
...
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality

minor

Location

contract.sol#L551,604

Description

Constant state variables should be declared constant to save gas.

```
swapAndLiquifyEnabled  
_tTotal
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality

minor

Location

contract.sol#L291,565,570,589,548,562,563,599

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.

```
PancakeSwapV2Router  
_symbol  
_name  
_decimals  
valuesFromGetValues  
balances  
feeRatesStruct  
WETH
```

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#L235,172,176,184,192,220,224,205,209,157 and 9 more

Description

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

```
values
length
contains
at
_values
_length
_at
_msgData
sendValue
...
```

Recommendation

Remove unused functions.

L07 - Missing Events Arithmetic

Criticality

minor

Location

contract.sol#L640

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.

```
numTokensToSwap = top * 10 ** _decimals
```

Recommendation

Emit an event for critical parameter changes.

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			
	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		
Address	Library			
	isContract	Internal		

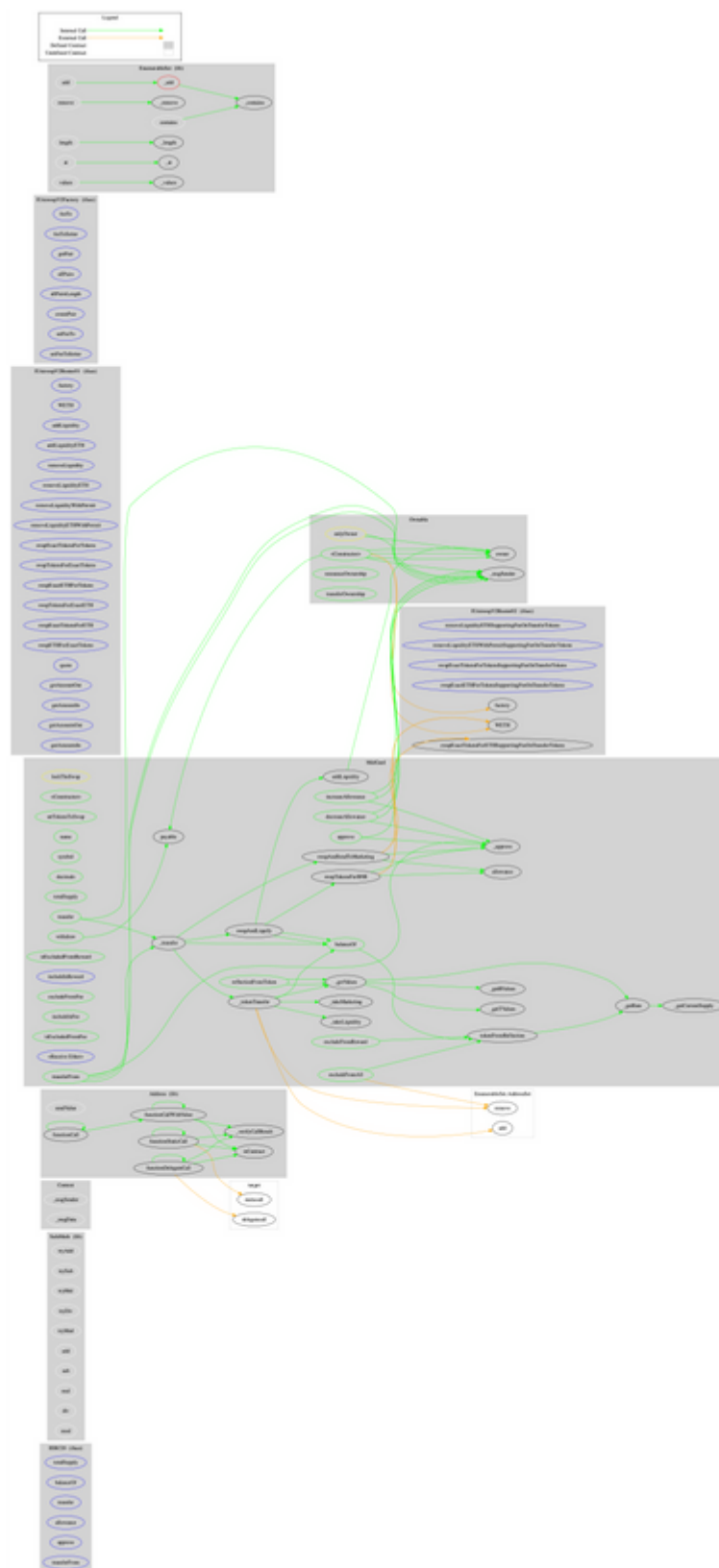
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	_verifyCallResult	Private		
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
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	✓	-
IUniswapV2Factory	Interface			
	feeTo	External		-
	feeToSetter	External		-
	getPair	External		-
	allPairs	External		-
	allPairsLength	External		-
	createPair	External	✓	-
	setFeeTo	External	✓	-
	setFeeToSetter	External	✓	-
EnumerableSet	Library			
	_add	Private	✓	
	_remove	Private	✓	
	_contains	Private		
	_length	Private		
	_at	Private		
	_values	Private		
	add	Internal	✓	
	remove	Internal	✓	
	contains	Internal		

	length	Internal		
	at	Internal		
	values	Internal		
MidGard	Implementation	Context, IERC20, Ownable		
	<Constructor>	Public	✓	-
	setTokensToSwap	Public	✓	onlyOwner
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	isExcludedFromReward	Public		-
	reflectionFromToken	Public		-
	tokenFromReflection	Public		-
	excludeFromReward	Public	✓	onlyOwner
	excludeFromAll	Public	✓	onlyOwner
	includeInReward	External	✓	onlyOwner
	excludeFromFee	Public	✓	onlyOwner
	includeInFee	Public	✓	onlyOwner
	isExcludedFromFee	Public		-
	<Receive Ether>	External	Payable	-
	_getValues	Private		
	_getTValues	Private		
	_getRValues	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	_takeMarketing	Private	✓	

	_takeLiquidity	Private	✓	
	_approve	Private	✓	
	_transfer	Private	✓	
	_tokenTransfer	Private	✓	
	swapAndLiquify	Private	✓	lockTheSwap
	swapAndSendToMarketing	Private	✓	lockTheSwap
	swapTokensForBNB	Private	✓	
	addLiquidity	Private	✓	
	withdraw	Public	✓	onlyOwner

Contract Flow



Domain Info

Domain Name	mid-gard-online.com
Registry Domain ID	2670992219_DOMAIN_COM-VRSN
Creation Date	2022-01-26T11:28:31Z
Updated Date	2022-01-26T11:28:32Z
Registry Expiry Date	
Registrar WHOIS Server	whois.publicdomainregistry.com
Registrar URL	www.publicdomainregistry.com
Registrar	PDR Ltd. d/b/a PublicDomainRegistry.com
Registrar IANA ID	303

The domain has been created about 1 month before the creation of the audit.

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

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 fixed value of 5% fees on buys and 7% fees on sales. All the tokens that are accumulated from the fees are liquidated and transferred to the dev's wallet.

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>