Cyberscope

Audit Report

World 6 Game

September 2022

Type	BEP20
Network	BSC
Address	0xC447424184d964537fA9E0D7B2d658b60DF289B3
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	World_6_Game
Compiler Version	v0.8.7+commit.e28d00a7
Optimization	200 runs
Licence	MIT
Explorer	https://bscscan.com/token/0xC447424184d964537fA9E0D7B2d658b60DF289B3
Symbol	W6G
Decimals	18
Total Supply	50,000,000
Domain	world6game.com

Audit Updates

Initial Audit	12th September 2022 https://github.com/cyberscope-io/audits/blob/main/w6g/v1/audit.pdf
Corrected	15th September 2022

Source Files

Filename	SHA256
Authorized.sol	de019cc6f2a52b295fef0221aff0ce3ca8a5fc803e64528243455da11ea5642c
Context.sol	91bb853b4716bea8540722c7b13af34e2f24b01a3654caddd68246a72e6c759b8
ERC20.sol	adea8d6813eba55020be787c8a9bef8a71f90c96fd5c75544bc4ac1bccbd51ab
IERC20.sol	4ad9e1338842c0a911ed1994774827c17eea117751a56f1111193d0bc4c0006e
IERC20Metadata.sol	aa7bbf621cc23ca80abde64c64ff6f9503aceae5592f8e3ed2d0ab0a345e09e6
IPancake.sol	ce5cedde1004c8768e88974c30f9d386f0e4f56f084d03e1397a5c8a2a274aa9
Ownable.sol	4a0c4ca403220345b36b1a2e02b5fe041c0e93aa5603a964344c4b3f2c3a8a36
Strings.sol	958c2a94731d0ceaaa0830b457842ec11471197f894c2afd6facd7709275f8ac
SwapHelper.sol	170476456e4c0725ddda16b1aa72b13289c108e7f01ac0ed11c0033dc89045ff
W6G.sol	9c5f0d2b1385f7399cdc15efbc35842686a54319f23b127d424cd5b6d69b4617

Solidity Assembly MethodId Analysis

MethodId	Method Name
0x70a08231	balanceOf(address)
0x022c0d9f	swap(uint256, uint256, address, bytes)
0x23b872dd	transferFrom(address, address, uint256)
0xa9059cbb	transfer(address, uint256)
0x0dfe1681	token0( )
0x0902f1ac	getReserves( )

Contract Analysis

● Critical ● Medium ● Minor / Informative ● Pass

Severity	Code	Description	Status
●	ST	Stops Transactions	Passed
●	OCTD	Transfers Contract's Tokens	Passed
●	OTUT	Transfers User's Tokens	Passed
●	ELFM	Exceeds Fees Limit	Passed
●	ULTW	Transfers Liquidity to Team Wallet	Unresolved
●	MT	Mints Tokens	Passed
●	BT	Burns Tokens	Passed
●	BC	Blacklists Addresses	Passed

ULTW - Transfers Liquidity to Team Wallet

Criticality	minor / informative
Location	contract.sol#L195
Status	Unresolved

Description

The contract owner has the authority to liquidate funds. These funds have been accumulated from fees collected from the contract. The liquidity can be transferred to a wallet by calling the buyBackAndHoldWithDecimals method.

```
  function buyBackWithDecimals(uint256 decimalAmount, address destAddress) private {
    -
    -
    -
    if (destAddress == address(0)) {
      swapToken(pairWbnbToken, reversed ? tokenAmount : 0, reversed ? 0 : tokenAmount,
swapHelperAddress);
      _burn(swapHelperAddress, tokenAmount);
      totalBurned += tokenAmount;
    } else {
      swapToken(pairWbnbToken, reversed ? tokenAmount : 0, reversed ? 0 : tokenAmount,
destAddress);
    }
    exemptFee[WBNB_TOKEN_PAIR] = previousExemptFeeState;
  }
```

Recommendation

The contract could embody a check for the maximum amount of funds that can be swapped. Since a huge amount may volatile the token's price.

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 / Informative

Severity	Code	Description	Status
●	PTFE	Paired Token Fees Exempt	Unresolved
●	MFEA	Misleading Fees Exempt Assumption	Unresolved
●	SAD	Swapped Amount Diversion	Unresolved
●	L01	Public Function could be Declared External	Unresolved
●	L02	State Variables could be Declared Constant	Unresolved
●	L04	Conformance to Solidity Naming Conventions	Unresolved
●	L05	Unused State Variable	Unresolved

PTFE - Paired Token Fees Exempt

Criticality	minor / informative
Location	contract.sol#L155
Status	Unresolved

Description

The contract assumes that during the swap of the BNB/Token pair, the contract should not tax the transfer. That means that the pairWbnbToken address should not be excluded from the exemptFee structure.

```
uint256 wbnbAmount = getAmountOut(feeTokenAmount, reserve1, reserve0);
    swapToken(pairWbnbToken, reversed ? 0 : wbnbAmount, reversed ? wbnbAmount : 0,
swapHelperAddress);
    uint256 wbnbBalanceNew = getTokenBalanceOf(wbnbAddress, swapHelperAddress);
    require(wbnbBalanceNew == wbnbBalanceBefore + wbnbAmount, "Wrong amount of
swapped on WBNB");
```

Recommendation

The contract should not allow the address pairWbnbToken to be removed from the exemptFee.

MFEA - Misleading Fees Exempt Assumption

Criticality	minor / informative
Location	contract.sol#L157,168
Status	Unresolved

Description

The contract is based on the fact that the external tokens WBNB and BUSD will never add fees in their transfer functionality.

```
    uint256 wbnbAmount = getAmountOut(feeTokenAmount, reserve1, reserve0);
    swapToken(pairWbnbToken, reversed ? 0 : wbnbAmount, reversed ? wbnbAmount : 0,
swapHelperAddress);
    uint256 wbnbBalanceNew = getTokenBalanceOf(wbnbAddress, swapHelperAddress);
    require(wbnbBalanceNew == wbnbBalanceBefore + wbnbAmount, "Wrong amount of
swapped on WBNB");

uint256 busdBalanceBefore = getTokenBalanceOf(busdAddress, address(this));
      tokenTransferFrom(wbnbAddress, swapHelperAddress, pairWbnbBusd, wbnbAmount);
      uint256 busdAmount = getAmountOut(wbnbAmount, reserve0, reserve1);
      swapToken(pairWbnbBusd, reversed ? busdAmount : 0, reversed ? 0 : busdAmount,
address(this));
      uint256 busdBalanceNew = getTokenBalanceOf(busdAddress, address(this));
      require(busdBalanceNew == busdBalanceBefore + busdAmount, "Wrong amount swapped
on BUSD");
```

Recommendation

Since it is an external factor that can be changed, the implementation could be more tolerant.

SAD - Swapped Amount Diversion

Criticality	minor
Location	contract.sol#L216
Status	Unresolved

Description

The _burn function should take into consideration the tokens that have been swapped and not the fixed number.

```
    tokenTransfer(WBNB, pairWbnbToken, wbnbAmount);

    uint256 tokenAmount = getAmountOut(wbnbAmount, reserve0, reserve1);
    if (destAddress == address(0)) {
      swapToken(pairWbnbToken, reversed ? tokenAmount : 0, reversed ? 0 : tokenAmount,
swapHelperAddress);
      _burn(swapHelperAddress, tokenAmount);
      totalBurned += tokenAmount;
    } else {
      swapToken(pairWbnbToken, reversed ? tokenAmount : 0, reversed ? 0 : tokenAmount,
destAddress);
    }
    exemptFee[WBNB_TOKEN_PAIR] = previousExemptFeeState;
```

Recommendation

The team is advised to carefully check if the implementation follows the expected business logic.

L01 - Public Function could be Declared External

Criticality	minor / informative
Location	ERC20.sol#L61,69,86,93,100,112,120,131,149,177,196 Ownable.sol#L53,61 W6G.sol#L66,67,70,71
Status	Unresolved

Description

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

```
name
symbol
decimals
totalSupply
balanceOf
transfer
allowance
approve
transferFrom
...
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality	minor / informative
Location	W6G.sol#L30,29,35,36
Status	Unresolved

Description

Constant state variables should be declared constant to save gas.

```
_maxAccountAmount
_maxTxAmount
feeDevelopmentWallet
feeStakePool1
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor / informative
Location	ERC20.sol#L35 W6G.sol#L11,321,322,323,324,21,22,25,26,27,29,30,51,53
Status	Unresolved

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.

```
_balances
World_6_Game
setWBNB_TOKEN_PAIR
setWBNB_BUSD_Pair
getWBNB_TOKEN_PAIR
getWBNB_BUSD_Pair
_name
_symbol
decimal
...
```

Recommendation

Follow the Solidity naming convention.
https://docs.soliditylang.org/en/v0.4.25/style-guide.html#naming-conventions.

L05 - Unused State Variable

Criticality	minor / informative
Location	W6G.sol#L13
Status	Unresolved

Description

There are segments that contain unused state variables.

```
ZERO
```

Recommendation

Remove unused state variables.

L09 - Dead Code Elimination

Criticality	minor / informative
Location	ERC20.sol#L220 Strings.sol#L39
Status	Unresolved

Description

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

```
_transfer
toHexString
```

Recommendation

Remove unused functions.

Contract Functions

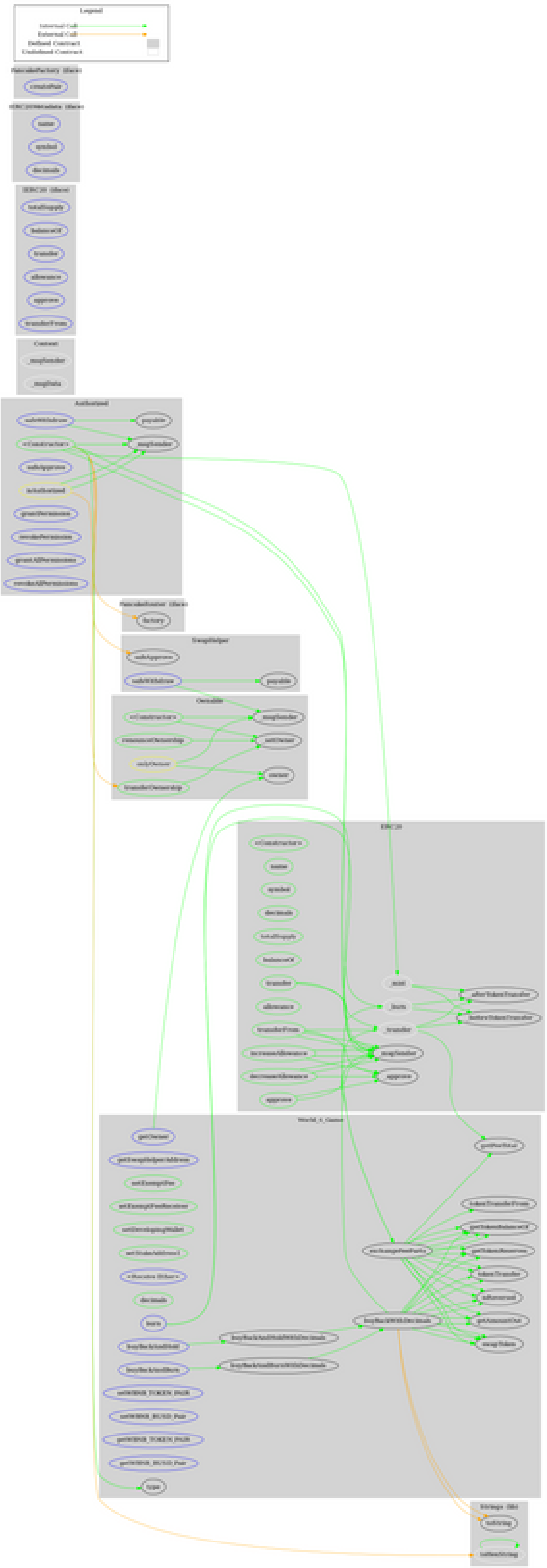
Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
Authorized	Implementation	Ownable		
	<Constructor>	Public	✓	-
	safeApprove	External	✓	isAuthorized
	safeWithdraw	External	✓	isAuthorized
	grantPermission	External	✓	isAuthorized
	revokePermission	External	✓	isAuthorized
	grantAllPermissions	External	✓	isAuthorized
	revokeAllPermissions	External	✓	isAuthorized
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
ERC20	Implementation	Context, IERC20, IERC20Metadata		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	Public	✓	-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	_transfer	Internal	✓	

	_mint	Internal	✓	
	_burn	Internal	✓	
	_approve	Internal	✓	
	_beforeTokenTransfer	Internal	✓	
	_afterTokenTransfer	Internal	✓	
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
IERC20Metadata	Interface	IERC20		
	name	External		-
	symbol	External		-
	decimals	External		-
PancakeFactory	Interface			
	createPair	External	✓	-
PancakeRouter	Interface			
	factory	External		-
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_setOwner	Private	✓	
Strings	Library			

	toString	Internal		
	toHexString	Internal		
	toHexString	Internal		
SwapHelper	Implementation	Ownable		
	<Constructor>	Public	✓	-
	safeApprove	External	✓	onlyOwner
	safeWithdraw	External	✓	onlyOwner
World_6_Game	Implementation	Authorized, ERC20		
	getOwner	External		-
	getFeeTotal	Public		-
	getSwapHelperAddress	External		-
	setExemptFee	Public	✓	onlyOwner
	setExemptFeeReceiver	Public	✓	onlyOwner
	setDevelopingWallet	Public	✓	onlyOwner
	setStakeAddress1	Public	✓	onlyOwner
	<Receive Ether>	External	Payable	-
	<Constructor>	Public	✓	ERC20
	decimals	Public		-
	_transfer	Internal	✓	
	exchangeFeeParts	Private	✓	
	burn	External	✓	-
	buyBackAndHold	External	✓	onlyOwner
	buyBackAndHoldWithDecimals	Public	✓	onlyOwner
	buyBackAndBurn	External	✓	onlyOwner
	buyBackAndBurnWithDecimals	Public	✓	onlyOwner
	buyBackWithDecimals	Private	✓	
	getAmountOut	Internal		
	isReversed	Internal		
	tokenTransfer	Internal	✓	
	tokenTransferFrom	Internal	✓	
	swapToken	Internal	✓	
	getTokenBalanceOf	Internal		
	getTokenReserves	Internal		

	setWBNB_TOKEN_PAIR	External	✓	onlyOwner
	setWBNB_BUSD_Pair	External	✓	onlyOwner
	getWBNB_TOKEN_PAIR	External		-
	getWBNB_BUSD_Pair	External		-

Contract Flow



Domain Info

Domain Name	world6game.com
Registry Domain ID	2647902325_DOMAIN_COM-VRSN
Creation Date	2021-10-15T00:37:43Z
Updated Date	2021-10-15T02:39:17Z
Registry Expiry Date	2022-10-15T00:37:43Z
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 was created 11 months before the creation of the audit. It will expire in about 1 month.

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

Summary

The Smart Contract analysis reported one minor severity issue. The contract owner has the authority to transfer funds to the team's wallet. Other than that, 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 fee of 7%.

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 provide all the essential tools to assist users draw their own conclusions.

The Cyberscope team

https://www.cyberscope.io