Cyberscope

Audit Report

Figures

March 2022

Commit fe02cfa119510fe6eabd5f75f3316a4ff3c08376

Github https://github.com/figurestoken/Contracts

Audited by © cyberscope

Table of Contents

Contract Review

Contract Name	Figures
Github	https://github.com/figurestoken/Contracts
Commit	fe02cfa119510fe6eabd5f75f3316a4ff3c08376
Symbol	FIGURES
Decimals	6
Total Supply	42,000,000
Domain	https://figures.exchange/

Source Files

Filename	SHA256
@chainlink/contracts/src/v0.8/interfaces/LinkTokenInterface.sol	c616881f74a1e5e584fefa0a97a22bcbcf92eb184e665a8384883d9590a19e89
@chainlink/contracts/src/v0.8/interfaces/VRFCoordinatorV2Interface.sol	1e120dcde1dc1cd1c5fe98e96e3c2cfb245e02b5d5e685a3216f553c9dbc9ed4
@chainlink/contracts/src/v0.8/VRFConsumerBaseV2.sol	32f174b1cb0f4becddf4a1f53b71710e6cd5165a4d2865ca17f805909211c258
@openzeppelin/contracts/access/Ownable.sol	75e3c97011e75627ffb36f4a2799a4e887e1a3e27ed427490e82d7b6f51cc5c9
@openzeppelin/contracts/token/ERC20/IERC20.sol	c2b06bb4572bb4f84bfc5477dadc0fcc497cb66c3a1bd53480e68bedc2e154a6

@openzeppelin/contracts/utils/Address.sol	aafa8f3e41700a8353aabcdf020e06735753e6bc4b615279b43de53cfbb4f2cd
@openzeppelin/contracts/utils/Context.sol	1458c260d010a08e4c20a4a517882259a23a4baa0b5bd9add9fb6d6a1549814a
@openzeppelin/contracts/utils/math/SafeMath.sol	15941f3904992a62ed117e93d9e2d5c4c22bd09a7ff97fdd5f49273cf09703ac
contracts/Figures.sol	14789e99c04d45e594f3297eb8e2c3e6886c555a8163dcb4de2e35378a673fde
contracts/IUni.sol	43b396437b445bbb515b4cfdb4413260f79aaa614b5ae2a05ce8b294b5466389

Audit Updates

Initial Audit	20th March 2022
Corrected	29th March 2022

Lottery Feature

The contact implements a lottery feature. All the users that buy or sell the token are getting a ticket. Every 7 days and 5 minutes 3 winners are shared the awarded amount. The awarded amount is USDT that has accumulated from the fees.

First Winner	Second Winner	Third Winner
80%	15%	5%

Notes about the lottery feature:

- The lottery duration is 7 days and 5 minutes.
- There are 3 winners that share the awarded amount proportionally to their position.
- The ticker randomization is using the Chainlink VRF mechanism that guarantees a decent distribution https://docs.chain.link/docs/chainlink-vrf/
- The winners can claim their rewards until 3 months after their declaration. Then, the contract owner can transfer the unclaimed prize back to the current jackpot minus a fee of 10%.
- The wallets that hold less than 0,000010 tokens are excluded from the winners even if they won.
- Each address can have up to 15 tickets.
- If the participants of the current round are less than 3, then the lottery process is extended for another 7 days and 5 minutes.

Recommendation:

The winners even if they were chosen, they may not be eligible to win. This will happen if they hold less than 0,000010 tokens. The duplicate winners will also be excluded. The lottery could proceed to the next winner in the queue, like a fourth randomly picked, rather than moving the corresponding funds to the marketing wallet.

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

Contract Diagnostics

Critical Medium Minor

Severity	Code	Description
Medium	FSA	Fixed Swap Address
Minor	MC	Missing Check
Minor	L01	Public Function could be Declared External
Minor	L02	State Variables could be Declared Constant
Minor	L04	Conformance to Solidity Naming Conventions

FSA - Fixed Swap Address

Criticality	minor
Location	contract.sol#L179

Description

The swap address is assigned once in the constructor and it can not be changed. The decentralized swaps sometimes create a new swap version or abandon the current. A contract that cannot change the swap address may not be able to catch-up the upgrade.

```
uniswapV2Router =
IUniswapV2Router02(0x7a250d5630B4cF539739dF2C5dAcb4c659F2488D);
```

Recommendation

It could be better to allow the swap address mutation in case of future swap updates.

MC - Missing Check

Criticality	medium
Location	contract.sol#L329

Description

The _transferFromPrizePot function is an essential part of the lottery feature. The lottery algorithm uses this functionality in order to move the funds to the winners. The transfer() function does not guarantee that the amount has been moved. Hence, if the transfer fails, the lottery functionality will not be interrupted and it will assume that the user has received the corresponding amount.

```
function  _transferFromPrizePot(uint256 _amountToSend, address to) private {
  _usdtToken.transfer(to, _amountToSend);
}
```

Recommendation

The contract could embed a safe transfer technique that will guarantee that the function will revert in case of a failure.

L01 - Public Function could be Declared External

Criticality	minor
Location	contracts/Figures.sol#L579,583,587,591

Description

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

```
totalSupply
decimals
symbol
name
```

Recommendation

Use the external attribute for functions never called from the contract

L02 - State Variables could be Declared Constant

Criticality	minor
Location	contracts/Figures.sol#L30,29,28,33,31,24,27

Description

Constant state variables should be declared constant to save gas.

```
vrfCoordinator
s_subscriptionId
requestConfirmations
numWords
link
keyHash
callbackGasLimit
```

Recommendation

Add the constant attribute to state variables that never change.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor
Location	contracts/Figures.sol#L20,21,24,34

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.

```
s_requestId
s_subscriptionId
LINKTOKEN
COORDINATOR
```

Recommendation

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

Contract Functions

Contract	Type	Bases		
	Function Name	Visibility	Mutability	Modifiers
LinkTokenInterface	Interface			
	allowance	External		-
	approve	External	✓	-
	balanceOf	External		-
	decimals	External		-
	decreaseApproval	External	✓	-
	increaseApproval	External	✓	-
	name	External		-
	symbol	External		-
	totalSupply	External		-
	transfer	External	✓	-
	transferAndCall	External	✓	-
	transferFrom	External	✓	-
VRFCoordinatorV2Interface	Interface			
	getRequestConfig	External		-
	requestRandomWords	External	✓	-
	createSubscription	External	✓	-
	getSubscription	External		-
	requestSubscriptionOwnerTransfer	External	✓	-
	acceptSubscriptionOwnerTransfer	External	✓	-
	addConsumer	External	✓	-
	removeConsumer	External	✓	-
	cancelSubscription	External	✓	-
VRFConsumerBaseV2	Implementation			
	<Constructor>	Public	✓	-

	fulfillRandomWords	Internal	✓	
	rawFulfillRandomWords	External	✓	-
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_transferOwnership	Internal	✓	
IERC20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
Address	Library			
	isContract	Internal		
	sendValue	Internal	✓	
	functionCall	Internal	✓	
	functionCall	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionCallWithValue	Internal	✓	
	functionStaticCall	Internal		
	functionStaticCall	Internal		
	functionDelegateCall	Internal	✓	
	functionDelegateCall	Internal	✓	
	verifyCallResult	Internal		
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		

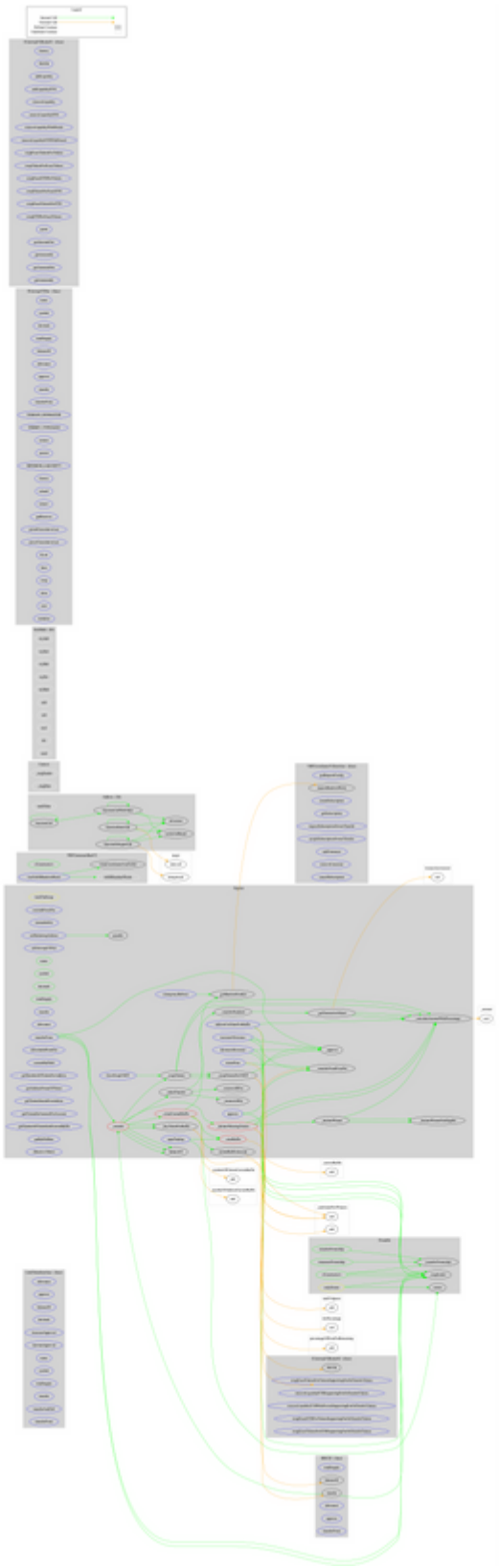
SafeMath	Library			
	tryAdd	Internal		
	trySub	Internal		
	tryMul	Internal		
	tryDiv	Internal		
	tryMod	Internal		
	add	Internal		
	sub	Internal		
	mul	Internal		
	div	Internal		
	mod	Internal		
	sub	Internal		
	div	Internal		
	mod	Internal		
Figures	Implementation	Context, IERC20, Ownable, VRFConsumerBaseV2		
	<Constructor>	Public	✓	VRFConsumerBaseV2
	_startRaffle	Private	✓	
	_closeCurrentRaffle	Private	✓	
	_getRandomNumber	Private	✓	
	fulfillRandomWords	Internal	✓	
	_declareWinningTickets	Private	✓	
	_declareWinner	Private	✓	
	_declareWinnerNotEligible	Private	✓	
	_transferFromPrizePot	Private	✓	
	_calculateAmountWithPercentage	Private		
	_removeAllFee	Private	✓	
	_restoreAllFee	Private	✓	
	_approve	Private	✓	
	_transfer	Private	✓	
	_buyTicketForRaffle	Private	✓	
	_swapTokens	Private	✓	lockTheSwap

	_swapTokensForUSDT	Private	✓	
	_tokenTransfer	Private	✓	
	_transferStandard	Private	✓	
	_getTransactionValues	Private		
	excludeFromFee	External	✓	onlyOwner
	includeInFee	External	✓	onlyOwner
	setMarketingAddress	External	✓	onlyOwner
	setUniswapV2Pair	External	✓	onlyOwner
	openTrading	External	✓	onlyOwner
	forceSwapUSDT	External	✓	onlyOwner
	redirectUnClaimForRaffle	External	✓	onlyOwner
	emergencyReDraw	External	✓	onlyOwner
	claimPrize	External	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
	increaseAllowance	External	✓	-
	decreaseAllowance	External	✓	-
	isExcludedFromFee	External		-
	currentRaffleId	External		-
	currentRaffleAmount	Public		-
	getNumbersOfTicketsForAddress	External		-
	getAddressOwnerOfTicket	External		-
	getTicketsDetailsForAddress	External		-
	getClaimableAmountForAccount	External		-
	getNumberOfTicketSoldCurrentRaffle	External		-
	getRaffleMeta	External		-
	<Receive Ether>	External	Payable	-

IUniswapV2Pair	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	✓	-
	MINIMUM_LIQUIDITY	External		-
	factory	External		-
	token0	External		-
	token1	External		-
	getReserves	External		-
	price0CumulativeLast	External		-
	price1CumulativeLast	External		-
	kLast	External		-
	burn	External	✓	-
	swap	External	✓	-
	skim	External	✓	-
	sync	External	✓	-
	initialize	External	✓	-
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	✓	-

Contract Flow



Domain Info

Domain Name	figures.exchange
Registry Domain ID	afe0f0f846684a17b52d33c14df5d470-DONUTS
Creation Date	2021-12-16T21:43:13Z
Updated Date	2021-12-21T21:43:55Z
Registry Expiry Date	2022-12-16T21:43:13Z
Registrar WHOIS Server	whois.namecheap.com
Registrar URL	https://www.namecheap.com/
Registrar	NameCheap, Inc.
Registrar IANA ID	1068

The domain has been created 3 months before the creation of the audit. It will expire in 9 months.

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 7% fee. The contract implements a lottery feature. The audit mentions some security concerns, business logic recommendations and performance improvements.

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