Cyberscope

Audit Report

Safe Universe

September 2022

Type	BEP20
Network	BSC
Address	0x44e5d6aa014ebf31d38ac5ed77393fd1851e1e3e
Audited by	© cyberscope

Table of Contents

Contract Review

Contract Name	SafeUniverse
Compiler Version	v0.8.8+commit.dddeac2f
Optimization	200 runs
Licence	Unlicense
Explorer	https://bscscan.com/token/0x44e5d6aa014ebf31d38ac5ed77393fd1851e1e3e
Symbol	SFU
Decimals	9
Total Supply	99,999,999,999,999,980,000
Domain	https://safeuniverse.io

Source Files

Filename	**SHA256**
contract.sol	e524097e41b9d06f99c375a0a8eb3ab9b6eb8693c72ab5c39deb2894187bb472

Audit Updates

Initial Audit	9th September 2022
Corrected	

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#L801
Status	Unresolved

Description

The contract owner has the authority to transfer funds without limit to the team wallet. These funds have been accumulated from fees collected from the contract. The owner may take advantage of it by calling the rescueBNB method.

```
function rescueBNB(uint256 weiAmount) external onlyOwner {
        require(address(this).balance >= weiAmount, "insufficient BNB balance");
        payable(msg.sender).transfer(weiAmount);
    }
```

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
●	STC	Succeeded Transfer Check	Unresolved
●	FSA	Fixed Swap Address	Unresolved
●	L01	Public Function could be Declared External	Unresolved
●	L02	State Variables could be Declared Constant	Unresolved
●	L03	Redundant Statements	Unresolved
●	L04	Conformance to Solidity Naming Conventions	Unresolved
●	L07	Missing Events Arithmetic	Unresolved
●	L13	Divide before Multiply Operation	Unresolved

STC - Succeeded Transfer Check

Criticality	minor / informative
Location	contract.sol#L809
Status	Unresolved

Description

According to the ERC20 specification, the transfer methods should be checked if the result is successful. Otherwise, the contract may wrongly assume that the transfer has been established.

```
function rescueAnyBEP20Tokens(address _tokenAddr,address _to, uint256 _amount)
public onlyOwner {
        require(_tokenAddr != address(this), "Owner can't claim contract's balance of its own
tokens");
        IBEP20(_tokenAddr).transfer(_to, _amount);
}
```

Recommendation

The contract should check if the result of the transfer methods is successful.

FSA - Fixed Swap Address

Criticality	minor / informative
Location	contract.sol#L212
Status	Unresolved

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.

```
constructor(address routerAddress) {
        IRouter _router = IRouter(routerAddress);
        address _pair = IFactory(_router.factory()).createPair(address(this), _router.WETH());

        router = _router;
        pair = _pair;
```

Recommendation

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

L01 - Public Function could be Declared External

Criticality	minor / informative
Location	contract.sol#L252,803,291,300,55,280,361,256,234,365,230,59,243,357,369,261,381,296,275
Status	Unresolved

Description

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

```
allowance
rescueAnyBEP20Tokens
transfer
reflectionFromToken
renounceOwnership
decreaseAllowance
includeInFee
approve
symbol
...
```

Recommendation

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

L02 - State Variables could be Declared Constant

Criticality	minor / informative
Location	contract.sol#L151,140
Status	Unresolved

Description

Constant state variables should be declared constant to save gas.

```
deadWallet
_tTotal
```

Recommendation

Add the constant attribute to state variables that never change.

L03 - Redundant Statements

Criticality	minor / informative
Location	contract.sol#L26
Status	Unresolved

Description

The contract contains statements that are not used and have no effect. As a result, those segments increase the code size of the contract unnecessarily.

```
Context
```

Recommendation

Remove the redundant statements in order to decrease the code size.

L04 - Conformance to Solidity Naming Conventions

Criticality	minor / informative
Location	contract.sol#L803,322,385,148,783,370,156,315,157,373,372,78,181,371,374,137,382,383,384,386
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.

```
_amount
_deadline
_dev
genesis_block
_enabled
_rfi
_name
EnableTrading
_symbol
...
```

Recommendation

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

L07 - Missing Events Arithmetic

Criticality	minor / informative
Location	contract.sol#L772,787,778,322
Status	Unresolved

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.

```
coolDownTime = time * 1
maxBuyLimit = maxBuy * 10 ** decimals()
swapTokensAtAmount = amount * 10 ** _decimals
deadline = _deadline
```

Recommendation

Emit an event for critical parameter changes.

L13 - Divide before Multiply Operation

Criticality	minor / informative
Location	contract.sol#L676
Status	Unresolved

Description

Performing divisions before multiplications may cause lose of prediction.

```
unitBalance = deltaBalance / (denominator - temp.liquidity)
```

Recommendation

The multiplications should be prior to the divisions.

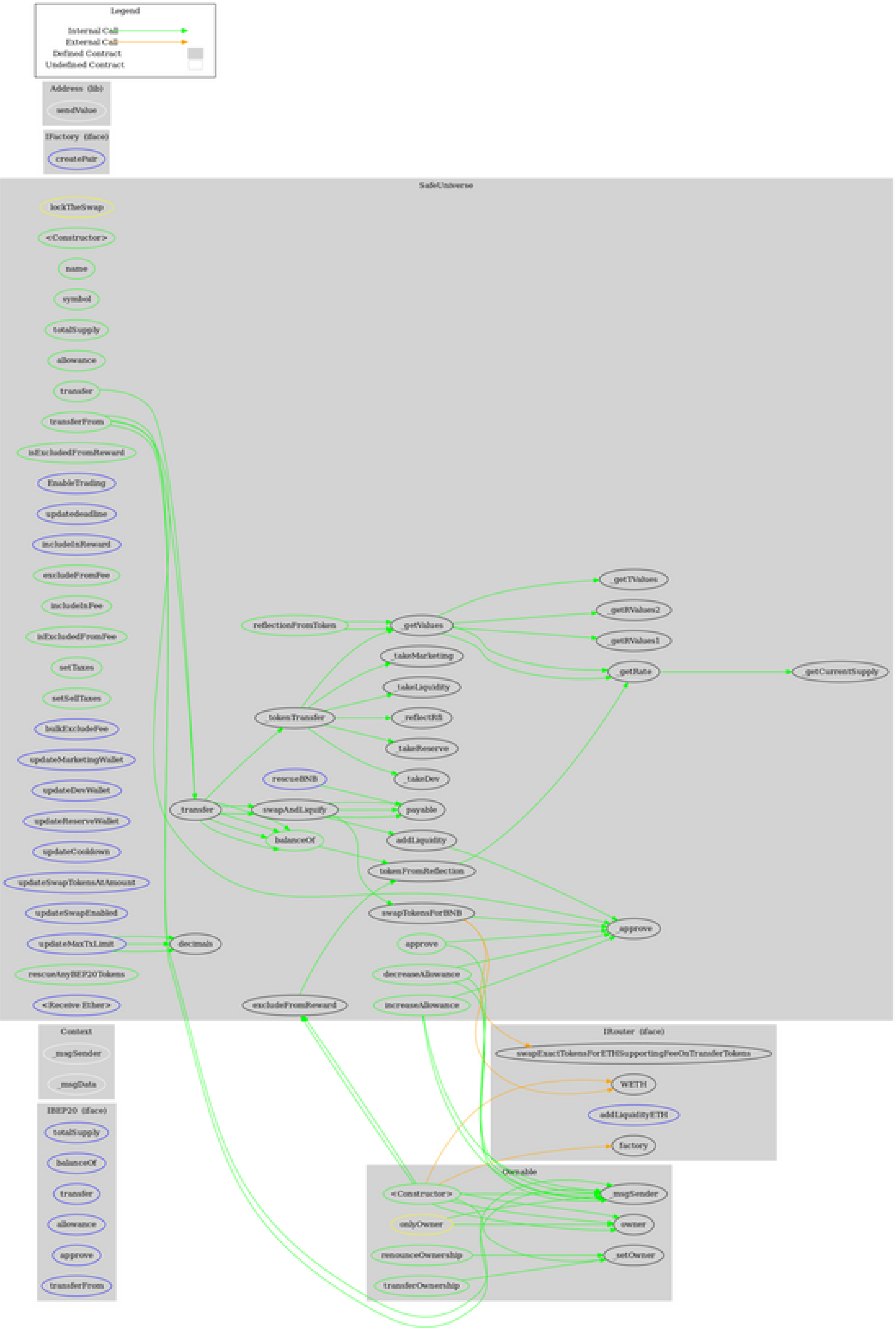
Contract Functions

Contract	**Type**	**Bases**		
	Function Name	**Visibility**	**Mutability**	**Modifiers**
IBEP20	Interface			
	totalSupply	External		-
	balanceOf	External		-
	transfer	External	✓	-
	allowance	External		-
	approve	External	✓	-
	transferFrom	External	✓	-
Context	Implementation			
	_msgSender	Internal		
	_msgData	Internal		
Ownable	Implementation	Context		
	<Constructor>	Public	✓	-
	owner	Public		-
	renounceOwnership	Public	✓	onlyOwner
	transferOwnership	Public	✓	onlyOwner
	_setOwner	Private	✓	
IFactory	Interface			
	createPair	External	✓	-
IRouter	Interface			
	factory	External		-
	WETH	External		-
	addLiquidityETH	External	Payable	-
	swapExactTokensForETHSupportingFeeOnTransferTokens	External	✓	-

Address	Library			
	sendValue	Internal	✓	
SafeUniverse	Implementation	Context, IBEP20, Ownable		
	<Constructor>	Public	✓	-
	name	Public		-
	symbol	Public		-
	decimals	Public		-
	totalSupply	Public		-
	balanceOf	Public		-
	allowance	Public		-
	approve	Public	✓	-
	transferFrom	Public	✓	-
	increaseAllowance	Public	✓	-
	decreaseAllowance	Public	✓	-
	transfer	Public	✓	-
	isExcludedFromReward	Public		-
	reflectionFromToken	Public		-
	EnableTrading	External	✓	onlyOwner
	updatedeadline	External	✓	onlyOwner
	tokenFromReflection	Public		-
	excludeFromReward	Public	✓	onlyOwner
	includeInReward	External	✓	onlyOwner
	excludeFromFee	Public	✓	onlyOwner
	includeInFee	Public	✓	onlyOwner
	isExcludedFromFee	Public		-
	setTaxes	Public	✓	onlyOwner
	setSellTaxes	Public	✓	onlyOwner
	_reflectRfi	Private	✓	
	_takeLiquidity	Private	✓	
	_takeMarketing	Private	✓	
	_takeDev	Private	✓	
	_takeReserve	Private	✓	

	_getValues	Private		
	_getTValues	Private		
	_getRValues1	Private		
	_getRValues2	Private		
	_getRate	Private		
	_getCurrentSupply	Private		
	_approve	Private	✓	
	_transfer	Private	✓	
	_tokenTransfer	Private	✓	
	swapAndLiquify	Private	✓	lockTheSwap
	addLiquidity	Private	✓	
	swapTokensForBNB	Private	✓	
	bulkExcludeFee	External	✓	onlyOwner
	updateMarketingWallet	External	✓	onlyOwner
	updateDevWallet	External	✓	onlyOwner
	updateReserveWallet	External	✓	onlyOwner
	updateCooldown	External	✓	onlyOwner
	updateSwapTokensAtAmount	External	✓	onlyOwner
	updateSwapEnabled	External	✓	onlyOwner
	updateMaxTxLimit	External	✓	onlyOwner
	rescueBNB	External	✓	onlyOwner
	rescueAnyBEP20Tokens	Public	✓	onlyOwner
	<Receive Ether>	External	Payable	-

Contract Flow



Domain Info

Domain Name	safeuniverse.io
Registry Domain ID	5fcdbed5871a47c798d0a041162b9942-DONUTS
Creation Date	2022-09-06T02:54:02Z
Updated Date	2022-09-06T02:54:05Z
Registry Expiry Date	2024-09-06T02:54:02Z
Registrar WHOIS Server	whois.namecheap.com
Registrar URL	https://www.namecheap.com/
Registrar	NameCheap, Inc.
Registrar IANA ID	1068

The domain was created 3 days before the creation of the audit. It will expire in almost 2 years.

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.

The contract has a throttling mechanism for sell transactions. The contract can pause sell transactions for up to 1 minutes for each user. In addition, on the first five blocks, there is a launch tax of 99%.

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 limit of max 12% fees.

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