Child Support Token

RugfreeCoins Verified on February 05th, 2024

Overview

✅ No mint function found, the owner cannot mint tokens after initial deployment.

✅ The owner can't set a max transaction limit

✅ The owner can't pause trading once it's enabled

❌ The owner must enable trade for the holders, if trading remains disabled, no one would be able to buy and sell.

✅ The owner can't change fees.

✅ The owner can't blacklist wallets.

✅ The owner can't set a max wallet limit

✅ The owner can't claim the contract's balance of its own token.

❗ **HIGH SEVERITY ISSUES**

The owner must enable trade for the holders, if trading remains disabled, no one would be able to buy and sell.

```
function tradingEnable() external onlyOwner {
    tradeEnable = true;
}
```

Contents

Audit details

Audited project
Child Support Token

Contract Address
0x530818A87573fB5a2cAa8CBBF89d8F3c5770d48E

Client contact
Child Support Token Team

Blockchain
Ethereum Mainnet

Project website
https://child.support/

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as at the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against us on the basis of what it says or doesn't say, or how we produced it, and it is important for you to conduct your own independent investigations before making any decisions. We go into more detail on this in the disclaimer below – **please make sure to read it in full.**

Background

RugfreeCoins was commissioned by the **Child Support Token Team** to perform an audit of the smart contract.

https://etherscan.io/token/0x530818A87573fB5a2cAa8CBBF89d8F3c5770d48E

This audit focuses on verifying that the smart contract is secure, resilient, and working according to the specifications.

The information in this report should be used to understand the risk exposure of the smart contract, project feasibility, and long-term sustainability, and as a guide to improving the smart contract's security posture by remediating the identified issues.

Tokenomics

▲ 5% tax when buying & selling

2% of trade goes to the marketing wallet in ETH
3% of trade goes to the holders in tokens

Target market and the concept

- Anyone who's interested in the Crypto space with long-term investment plans.
- Anyone who's ready to earn a passive income by holding tokens.
- Anyone who's interested in trading tokens.
- Anyone who's interested in taking part in the Child Support token ecosystem.
- Anyone who's interested in taking part in the future plans of Child Support Token.
- Anyone who's interested in making financial transactions with any other party Child Support Token as the currency.

Potential to grow with score points

⚡ Project efficiency	**10** / 10
🌟 Project uniqueness	**10** / 10
📊 Information quality	**10** / 10
💧 Service quality	**10** / 10
💻 System quality	**10** / 10
🌍 Impact on the community	**10** / 10
💼 Impact on the business	**10** / 10
🔮 Preparing for the future	**10** / 10
🔓 Smart contract security	**9** / 10
🛠️ Smart contract functionality assessment	**10** / 10
🏆 **Total Score**	**9.9** / 10

Contract details

Token contract details for 05th of February 2024

Contract name	Child Support
Contract address	0x530818A87573fB5a2cAa8CBBF89d8F3c5770d48E
Token supply	2,100,000,000
Token ticker	$CS
Decimals	9
Token holders	1
Transaction count	1
Contract deployer address	0x15B2734f5B23106520BB948DfE5E79828a8b6Ed3
Contract's current owner address	0x15B2734f5B23106520BB948DfE5E79828a8b6Ed3
Marketing wallet	0x835050495B3779116138d835Fc20C98e05152624

Contract code function details

№	Category	Item	Result
1	Coding conventions	ERC20 Token standards	PASS
		Compile errors	PASS
		Compiler version security	PASS
		Visibility specifiers	PASS
		Gas consumption	PASS
		SafeMath features	PASS
		Fallback usage	PASS
		tx.origin usage	PASS
		Deprecated items	PASS
		Redundant code	PASS
		Overriding variables	PASS
2	Function call audit	Authorization of function call	PASS
		Low level function (call/delegate call) security	PASS
		Returned value security	PASS
		Self destruct function security	PASS
3	Business security & centralisation	Access control of owners	HIGH
		Business logics	PASS
		Business implementation	PASS
4	Integer overflow/underflow		PASS
5	Reentrancy		PASS
6	Exceptional reachable state		PASS
7	Transaction ordering dependence		PASS
8	Block properties dependence		PASS
9	Pseudo random number generator (PRNG)		PASS
10	DoS (Denial of Service)		PASS
11	Token vesting implementation		PASS
12	Fake deposit		PASS
13	Event security		PASS

Contract description table

The below table represents the summary of the contracts and methods in the token contract. We scanned the whole contract and listed down all the Interfaces, functions, and implementations with their visibility and mutability.

Contract	Type	Bases		
L	Function Name	Visibility	Mutability	Modifiers
CS	Implementation	IERC20, Ownable		
L		Public ❗	🔴	NO ❗
L	name	Public ❗		NO ❗
L	symbol	Public ❗		NO ❗
L	decimals	Public ❗		NO ❗
L	totalSupply	Public ❗		NO ❗
L	balanceOf	Public ❗		NO ❗
L	transfer	Public ❗	🔴	NO ❗
L	allowance	Public ❗		NO ❗
L	approve	Public ❗	🔴	NO ❗
L	transferFrom	Public ❗	🔴	NO ❗
L	isExcludedFromReward	Public ❗		NO ❗
L	totalFees	Public ❗		NO ❗
L	deliver	Public ❗	🔴	NO ❗
L	reflectionFromToken	Public ❗		NO ❗
L	tokenFromReflection	Public ❗		NO ❗
L	excludeFromReward	Public ❗	🔴	onlyOwner
L	includeInReward	External ❗	🔴	onlyOwner
L	changeMarketingWallet	External ❗	🔴	onlyOwner

L	changeSwapAmount	External ❗	🛑	onlyOwner
L	setExcludedFromFee	External ❗	🛑	onlyOwner
L	tradingEnable	External ❗	🛑	onlyOwner
L	updateBuyFees	External ❗	🛑	onlyOwner
L	updateSellFees	External ❗	🛑	onlyOwner
L	updateTransferFees	External ❗	🛑	onlyOwner
L	setSwapAndLiquifyEnabled	Public ❗	🛑	onlyOwner
L		External ❗	💵	NO ❗
L	setAuthorizedWallets	External ❗	🛑	onlyOwner
L	setExcludedFromAutoLiquidity	External ❗	🛑	onlyOwner
L	_reflectFee	Private 🔐	🛑	
L	_getTValues	Private 🔐		
L	_getRValues	Private 🔐		
L	_getRate	Private 🔐		
L	_getCurrentSupply	Private 🔐		
L	takeTokenFees	Private 🔐	🛑	
L	takeTransactionFee	Private 🔐	🛑	
L	calculateFee	Private 🔐		
L	isExcludedFromFee	Public ❗		NO ❗
L	_approve	Private 🔐	🛑	

L	_transfer	Private 🔐	🛑	
L	swapAndSendETH	Private 🔐	🛑	lockTheSwap
L	swapTokensForEth	Private 🔐	🛑	
L	_tokenTransfer	Private 🔐	🛑	
L	_transferStandard	Private 🔐	🛑	
L	_transferBothExcluded	Private 🔐	🛑	
L	_transferToExcluded	Private 🔐	🛑	
L	_transferFromExcluded	Private 🔐	🛑	
L	toggleTransferFee	External ❗	🛑	onlyOwner

Ownable	**Implementation**	**Context**		
L		Public ❗	🛑	NO ❗
L	owner	Public ❗		NO ❗
L	_checkOwner	Internal 🔒		
L	renounceOwnership	Public ❗	🛑	onlyOwner
L	transferOwnership	Public ❗	🛑	onlyOwner
L	_transferOwnership	Internal 🔒	🛑	

IERC20	**Interface**			
L	totalSupply	External ❗		NO ❗
L	balanceOf	External ❗		NO ❗
L	transfer	External ❗	🛑	NO ❗

L	allowance	External ❗		NO ❗
L	approve	External ❗	🛑	NO ❗
L	transferFrom	External ❗	🛑	NO ❗
Context	**Implementation**			
L	_msgSender	Internal 🔒		
L	_msgData	Internal 🔒		
IUniswapV2 Factory	**Interface**			
L	feeTo	External ❗		NO ❗
L	feeToSetter	External ❗		NO ❗
L	getPair	External ❗		NO ❗
L	allPairs	External ❗		NO ❗
L	allPairsLength	External ❗		NO ❗
L	createPair	External ❗	🛑	NO ❗
L	setFeeTo	External ❗	🛑	NO ❗
L	setFeeToSetter	External ❗	🛑	NO ❗
IUniswapV2 Pair	**Interface**			
L	name	External ❗		NO ❗
L	symbol	External ❗		NO ❗

15

L	decimals	External ❗		NO ❗
L	totalSupply	External ❗		NO ❗
L	balanceOf	External ❗		NO ❗
L	allowance	External ❗		NO ❗
L	approve	External ❗	🔴	NO ❗
L	transfer	External ❗	🔴	NO ❗
L	transferFrom	External ❗	🔴	NO ❗
L	DOMAIN_SEPARATOR	External ❗		NO ❗
L	PERMIT_TYPEHASH	External ❗		NO ❗
L	nonces	External ❗		NO ❗
L	permit	External ❗	🔴	NO ❗
L	MINIMUM_LIQUIDITY	External ❗		NO ❗
L	factory	External ❗		NO ❗
L	token0	External ❗		NO ❗
L	token1	External ❗		NO ❗
L	getReserves	External ❗		NO ❗
L	price0CumulativeLast	External ❗		NO ❗
L	price1CumulativeLast	External ❗		NO ❗
L	kLast	External ❗		NO ❗

└	mint	External ❗	🛑	NO ❗
└	burn	External ❗	🛑	NO ❗
└	swap	External ❗	🛑	NO ❗
└	skim	External ❗	🛑	NO ❗
└	sync	External ❗	🛑	NO ❗
└	initialize	External ❗	🛑	NO ❗
IUniswapV2 Router01	**Interface**			
└	factory	External ❗		NO ❗
└	WETH	External ❗		NO ❗
└	addLiquidity	External ❗	🛑	NO ❗
└	addLiquidityETH	External ❗	💵	NO ❗
└	removeLiquidity	External ❗	🛑	NO ❗
└	removeLiquidityETH	External ❗	🛑	NO ❗
└	removeLiquidityWithPermit	External ❗	🛑	NO ❗
└	removeLiquidityETHWithPermit	External ❗	🛑	NO ❗
└	swapExactTokensForTokens	External ❗	🛑	NO ❗
└	swapTokensForExactTokens	External ❗	🛑	NO ❗
└	swapExactETHForTokens	External ❗	💵	NO ❗
└	swapTokensForExactETH	External ❗	🛑	NO ❗
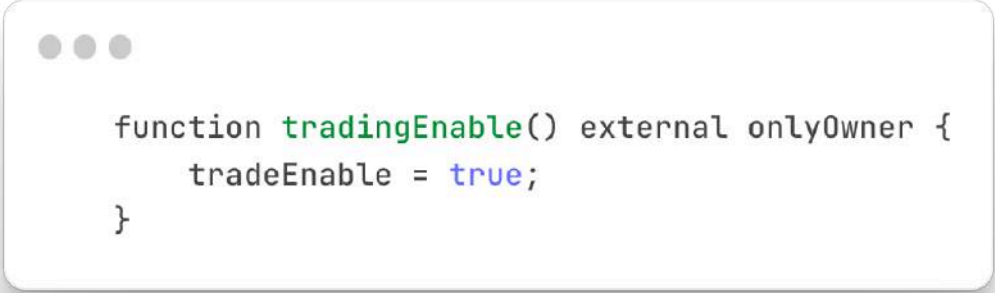
CS					
L	swapExactTokensForETH	External ❗	🛑		NO ❗
L	swapETHForExactTokens	External ❗	💵		NO ❗
L	quote	External ❗			NO ❗
L	getAmountOut	External ❗			NO ❗
L	getAmountIn	External ❗			NO ❗
L	getAmountsOut	External ❗			NO ❗
L	getAmountsIn	External ❗			NO ❗
CS	**Implementation**	**IERC20, Ownable**			
L		Public ❗	🛑		NO ❗
L	name	Public ❗			NO ❗
L	symbol	Public ❗			NO ❗
L	decimals	Public ❗			NO ❗
L	totalSupply	Public ❗			NO ❗
L	balanceOf	Public ❗			NO ❗
L	transfer	Public ❗	🛑		NO ❗
L	allowance	Public ❗			NO ❗
L	approve	Public ❗	🛑		NO ❗
L	transferFrom	Public ❗	🛑		NO ❗
L	isExcludedFromReward	Public ❗			NO ❗
L	totalFees	Public ❗			NO ❗
L	deliver	Public ❗	🛑		NO ❗
L	reflectionFromToken	Public ❗			NO ❗
L	tokenFromReflection	Public ❗			NO ❗
L	excludeFromReward	Public ❗	🛑		onlyOwner

L	includeInReward	External ❗	🛑	onlyOwner
L	changeMarketingWallet	External ❗	🛑	onlyOwner
L	changeSwapAmount	External ❗	🛑	onlyOwner
L	setExcludedFromFee	External ❗	🛑	onlyOwner
L	tradingEnable	External ❗	🛑	onlyOwner
L	updateBuyFees	External ❗	🛑	onlyOwner
L	updateSellFees	External ❗	🛑	onlyOwner
L	updateTransferFees	External ❗	🛑	onlyOwner
L	setSwapAndLiquifyEnabled	Public ❗	🛑	onlyOwner
L		External ❗	💵	NO ❗
L	setAuthorizedWallets	External ❗	🛑	onlyOwner
L	setExcludedFromAutoLiquidity	External ❗	🛑	onlyOwner
L	_reflectFee	Private 🔐	🛑	
L	_getTValues	Private 🔐		
L	_getRValues	Private 🔐		
L	_getRate	Private 🔐		
L	_getCurrentSupply	Private 🔐		
L	takeTokenFees	Private 🔐	🛑	
L	takeTransactionFee	Private 🔐	🛑	
L	calculateFee	Private 🔐		

L	isExcludedFromFee	Public ❗		NO ❗
L	_approve	Private 🔐	🛑	
L	_transfer	Private 🔐	🛑	
L	swapAndSendETH	Private 🔐	🛑	lockTheSwap
L	swapTokensForEth	Private 🔐	🛑	
L	_tokenTransfer	Private 🔐	🛑	
L	_transferStandard	Private 🔐	🛑	
L	_transferBothExcluded	Private 🔐	🛑	
L	_transferToExcluded	Private 🔐	🛑	
L	_transferFromExcluded	Private 🔐	🛑	
L	toggleTransferFee	External ❗	🛑	onlyOwner

Legend

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable

Inheritance Hierarchy

Security issue checking status

❖ High severity issues

The owner must enable trade for the holders, if trading remains disabled, no one would be able to buy and sell.

```
function tradingEnable() external onlyOwner {
    tradeEnable = true;
}
```

❖ Medium severity issues

No medium severity issues found

❖ Low severity issues

No high severity issues found

Owner privileges

❖ Owner can include/exclude wallets from rewards

```solidity
function excludeFromReward(address account) public onlyOwner {
    require(!_isExcluded[account], "Account is already excluded");

    if (_rOwned[account] > 0) {
        _tOwned[account] = tokenFromReflection(_rOwned[account]);
    }
    _isExcluded[account] = true;
    _excluded.push(account);
}

function includeInReward(address account) external onlyOwner {
    require(_isExcluded[account], "Account is already excluded");

    for (uint256 i = 0; i < _excluded.length; i++) {
        if (_excluded[i] == account) {
            _excluded[i] = _excluded[_excluded.length - 1];
            _tOwned[account] = 0;
            _isExcluded[account] = false;
            _excluded.pop();
            break;
        }
    }
}
```

❖ Owner can change marketing wallet

```solidity
function changeMarketingWallet(address _wallet) external onlyOwner {
    marketingWallet = _wallet;
}
```

❖ Owner can change swap point minimum up-to 1 token

```solidity
function changeSwapAmount(uint256 amount) external onlyOwner {
    _swapTokensAt = amount * 10 ** 9;
}
```

❖ Owner can include/exclude wallets from fees

```solidity
function setExcludedFromFee(address account, bool e) external onlyOwner {
    _isExcludedFromFee[account] = e;
}
```

❖ Owner can enable trading,once enabled can not disable again

```solidity
function tradingEnable() external onlyOwner {
    tradeEnable = true;
}
```

❖ Owner can change buy/sell and transfer fees up to 5% (each fees)

```solidity
function updateBuyFees(
    uint256 _RewardFee,
    uint256 _MarketingFee
) external onlyOwner {
    buyRewardFee = _RewardFee;
    buyMarketingFee = _MarketingFee;

    buyTotalFee = _RewardFee + _MarketingFee;
    require(buyTotalFee <= 5, "Fees can not greater than 5%");
}

function updateSellFees(
    uint256 _RewardFee,
    uint256 _MarketingFee
) external onlyOwner {
    sellRewardFee = _RewardFee;
    sellMarketingFee = _MarketingFee;

    sellTotalFee = _RewardFee + _MarketingFee;

    require(sellTotalFee <= 5, "Fees can not greater than 5%");
}

function updateTransferFees(
    uint256 _RewardFee,
    uint256 _MarketingFee
) external onlyOwner {
    transferlRewardFee = _RewardFee;
    transferMarketingFee = _MarketingFee;

    transferTotalFee = _RewardFee + _MarketingFee;

    require(transferTotalFee <= 5, "Fees can not greater than 5%");
}
```

❖ Owner can enable/disable swapping

```solidity
function setSwapAndLiquifyEnabled(bool e) public onlyOwner {
    _swapAndLiquifyEnabled = e;
    emit SwapAndLiquifyEnabledUpdated(e);
}
```

❖ Owner can add/remove authorized wallets

```solidity
function setAuthorizedWallets(
    address wallet,
    bool status
) external onlyOwner {
    _isAuthorized[wallet] = status;
}
```

Audit conclusion

RugFreeCoins team has performed in-depth testing, line-by-line manual code review, and automated audit of the smart contract. The smart contract was analyzed mainly for common smart contract vulnerabilities, exploits, manipulations, and hacks. According to the smart contract audit.

Smart contract functional Status:	PASS ▾
Smart contract security Status:	HIGH ISSUES ▾
Number of risk issues:	01
Solidity code functional issue level:	PASS ▾
Number of owner privileges:	08
Centralization risk correlated to the active owner:	HIGH ▾
Smart contract active ownership:	ACTIVE ▾