Coinsult

Advanced Manual Smart Contract Audit

Project: SET SPACE TOKEN
Website: https://setspacetoken.com/

● **Low-risk**	● **Medium-risk**	● **High-risk**
5 low-risk code issues found	0 medium-risk code issues found	0 high-risk code issues found

Contract address

0x8ae69eaef5ae4860f7b16b85ab50b3672ddd2da4

Disclaimer: Coinsult is not responsible for any financial losses. Nothing in this contract audit is financial advice, please do your own research.

Disclaimer

Coinsult is not responsible if a project turns out to be a scam, rug-pull or honeypot. We only provide a detailed analysis for your own research.

Coinsult is not responsible for any financial losses. Nothing in this contract audit is financial advice, please do your own research.

The information provided in this audit is for informational purposes only and should not be considered investment advice. Coinsult does not endorse, recommend, support or suggest to invest in any project.

Coinsult can not be held responsible for when a project turns out to be a rug-pull, honeypot or scam.

Tokenomics

Rank	Address	Quantity (Token)	Percentage
1	Null Address: 0x000...dEaD	100,000,000,000,000,000	50.0000%
2	0x63d357ad2ae6fe4450dd057b864c73aefaaa2869	50,000,000,000,000,000	25.0000%
3	0xd5f5503a85b39bdf250e82c1edefa73da047b955	30,000,000,000,000,000	15.0000%
4	0xe7c6c3d8c9ce070a369bad20e6ed6736239cfa8b	10,000,000,000,000,000	5.0000%
5	0xa1a8561529f11eef77b15ed390e2e133810f8096	10,000,000,000,000,000	5.0000%

Source code

Coinsult was commissioned by Set Space Token to perform an audit based on the following smart contract:

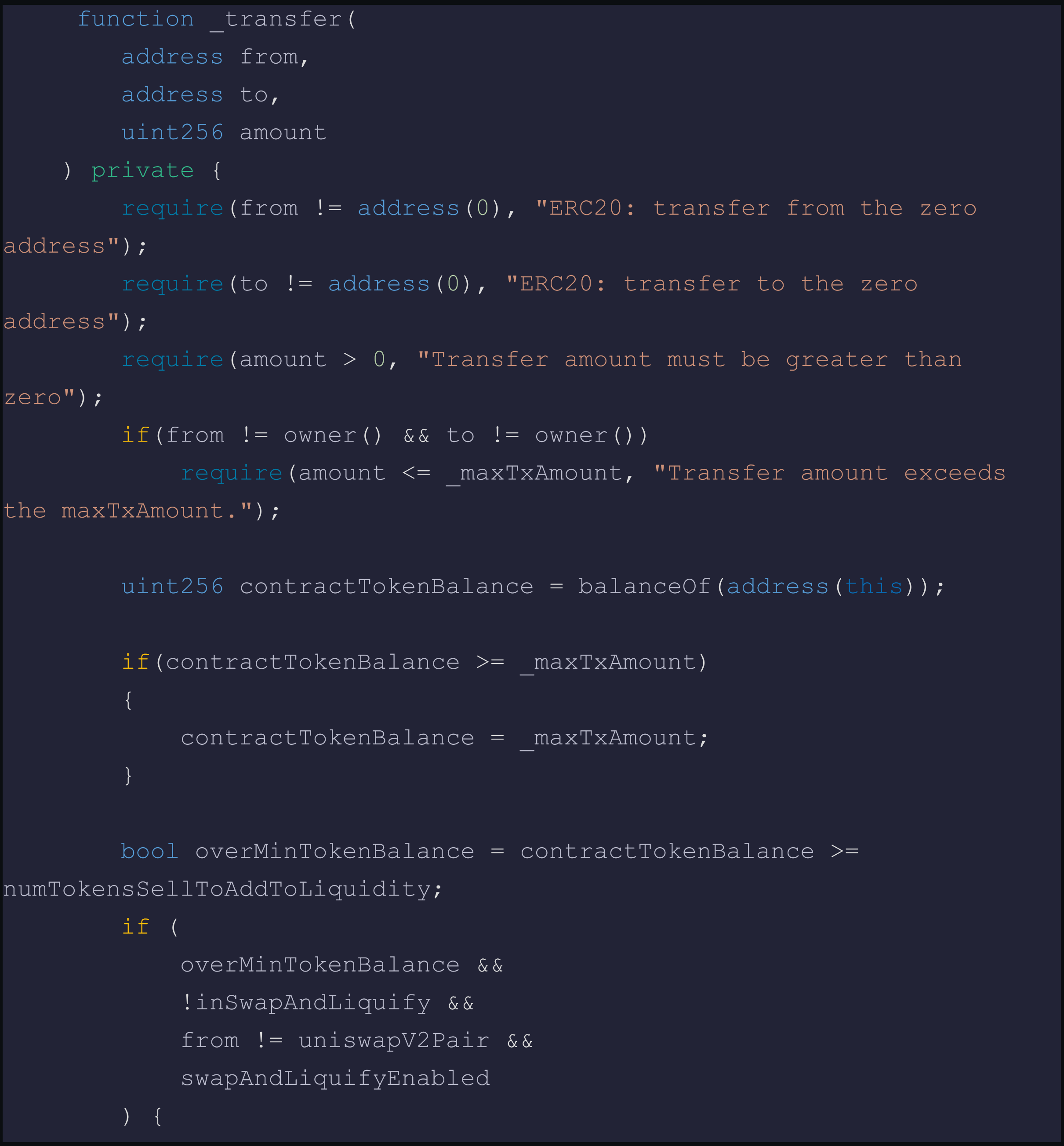
https://bscscan.com/address/0x8ae69eaef5ae4860f7b16b85ab50b3672ddd2da4#code

Manual Code Review

● Low-risk

5 low-risk code issues found.
Could be fixed, will not bring problems.

- Contract contains Reentrancy vulnerabilities:
  Additional information: This combination increases risk of malicious intent. While it may be justified by some complex mechanics (e.g. rebase, reflections, buyback).
  More information: Slither



```
     function _transfer(
        address from,
        address to,
        uint256 amount
    ) private {
        require(from != address(0), "ERC20: transfer from the zero
address");
        require(to != address(0), "ERC20: transfer to the zero
address");
        require(amount > 0, "Transfer amount must be greater than
zero");
        if(from != owner() && to != owner())
            require(amount <= _maxTxAmount, "Transfer amount exceeds
the maxTxAmount.");

        uint256 contractTokenBalance = balanceOf(address(this));

        if(contractTokenBalance >= _maxTxAmount)
        {
            contractTokenBalance = _maxTxAmount;
        }

        bool overMinTokenBalance = contractTokenBalance >=
numTokensSellToAddToLiquidity;
        if (
            overMinTokenBalance &&
            !inSwapAndLiquify &&
            from != uniswapV2Pair &&
            swapAndLiquifyEnabled
        ) {
            contractTokenBalance = numTokensSellToAddToLiquidity;
            swapAndLiquify(contractTokenBalance);
        }

        bool takeFee = true;
        if(_isExcludedFromFee[from] || _isExcludedFromFee[to]){
            takeFee = false;
        }

        _tokenTransfer(from,to,amount,takeFee);
    }
```

- Costly operations inside a loop might waste gas, so optimizations are justified.

  Use a local variable to hold the loop computation result.

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

- Avoid relying on block.timestamp

  block.timestamp can be manipulated by miners.

```
function unlock() public virtual {
        require(_previousOwner == msg.sender, "You don't have
permission to unlock.");
        require(block.timestamp > _lockTime , "Contract is locked.");
        emit OwnershipTransferred(_owner, _previousOwner);
        _owner = _previousOwner;
    }
}
```

- Missing zero address validation
  Check that the new address is not the zero address.

```
        _devWalletAddress = feeaddress;
```

- Redundant statement
  Remove statements if they congest code but offer no value (“this;”)

```
function _msgData() internal view virtual returns (bytes calldata) {
        this; // silence state mutability warning without generating
bytecode - see https://github.com/ethereum/solidity/issues/2691
        return msg.data;
    }
}
```

🟡 Medium-risk

0 medium-risk code issues found.
Should be fixed, could bring problems.

🔴 High-risk

0 high-risk code issues found
Must be fixed, and will bring problems.

Extra notes by the team

🟡 Owner can whitelist addresses from fees.

🟡 The ownership of the contract isn’t renounced.

🟡 Owner can set a max transaction amount.

🟡 Fees can be set up to 100% for both buy and sell fees.

🟡 Owner can change the router address

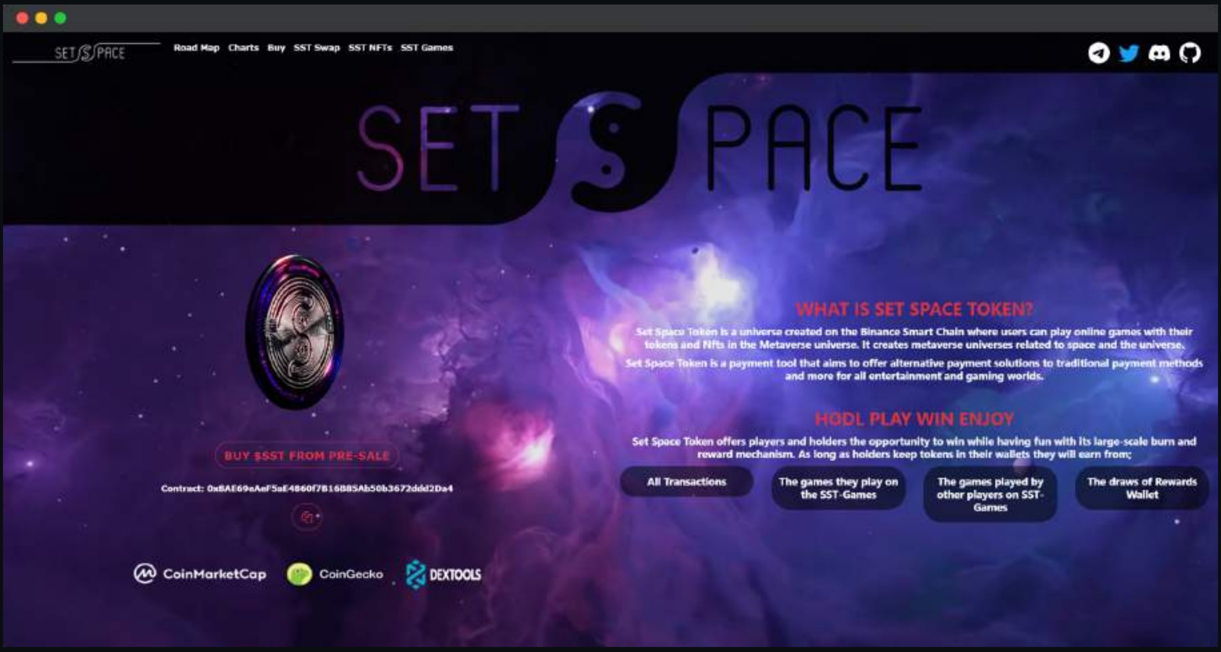
🔴 Owner is able to pause the contract

Contract Snapshot

```
contract CoinToken is Context, IERC20, Ownable {
    using SafeMath for uint256;
    using Address for address;

    mapping (address => uint256) private _rOwned;
    mapping (address => uint256) private _tOwned;
    mapping (address => mapping (address => uint256)) private
_allowances;
    mapping (address => bool) private _isExcludedFromFee;
    mapping (address => bool) private _isExcluded;
    address[] private _excluded;
    address public _devWalletAddress;     // TODO - team wallet here
    uint256 private constant MAX = ~uint256(0);
    uint256 private _tTotal;
    uint256 private _rTotal;
    uint256 private _tFeeTotal;
    string private _name;
    string private _symbol;
    uint256 private _decimals;
    uint256 public _taxFee;
    uint256 private _previousTaxFee;
    uint256 public _devFee;
    uint256 private _previousDevFee;
    uint256 public _liquidityFee;
    uint256 private _previousLiquidityFee;
    IUniswapV2Router02 public uniswapV2Router;
    address public uniswapV2Pair;
    bool inSwapAndLiquify;
    bool public swapAndLiquifyEnabled = true;
    uint256 public _maxTxAmount;
    uint256 public numTokensSellToAddToLiquidity;
    event MinTokensBeforeSwapUpdated(uint256 minTokensBeforeSwap);
    event SwapAndLiquifyEnabledUpdated(bool enabled);
    event SwapAndLiquify(
        uint256 tokensSwapped,
        uint256 ethReceived,
        uint256 tokensIntoLiqudity
    );
```

Website Review



WHAT IS SET SPACE TOKEN?

Set Space Token is a universe created on the Binance Smart Chain where users can play online games with their tokens and Nfts in the Metaverse universe. It creates metaverse universes related to space and the universe.

Set Space Token is a payment tool that aims to offer alternative payment solutions to traditional payment methods and more for all entertainment and gaming worlds.

HODL PLAY WIN ENJOY

Set Space Token offers players and holders the opportunity to win while having fun with its large-scale burn and reward mechanism. As long as holders keep tokens in their wallets they will earn from;

Contract: 0x8AE69eAeF5aE4860f7B16B85Ab50b3672ddd2Da4

Coinsult checks the website completely manually and looks for visual, technical and textual errors. We also look at the security, speed and accessibility of the website. In short, a complete check to see if the website meets the current standard of the web development industry.

🟢 Mobile Friendly
🟢 Contains no jQuery errors
🟢 SSL Secured
🟢 No major spelling errors

Loading speed: 82%

Rug-pull Review

Based on the available information analyzed by us, we come to the following conclusions:

🟢 Locked Liquidity (no liquidity yet)

🟡 Large unlocked wallets

- Note: Tokens can still be distributed before presale

🔴 No doxxed Team

Honeypot Review

Based on the available information analyzed by us, we come to the following conclusions:

🟢 Ability to sell

🔴 Owner is able to pause the contract

🟡 Router not hard coded in the contract

Note: Please check the disclaimer above and note, the audit makes no statements or warranties on business model, investment attractiveness or code sustainability. The report is provided for the only contract mentioned in the report and does not include any other potential contracts deployed by the project owner.