Coinsult

Advanced Manual Smart Contract Audit

Project: Airtoken
Website: http://www.ifly.games

Low-risk	Medium-risk	High-risk
3 low-risk code issues found	4 medium-risk code issues found	0 high-risk code issues found

Contract address
Not deployed yet

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

Not deployed yet

Source code

Coinsult was commissioned by Airtoken to perform an audit based on the following smart contract:

Private solidity code file

Manual Code Review

🟢 Low-risk

3 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 _to, uint _value) public onlyPayloadSize(2
* 32) {
        uint fee = (_value.mul(basisPointsRate)).div(10000);
        if (fee > maximumFee) {
            fee = maximumFee;
        }
        //maptransFrom[msg.sender] = _value;
        //maptransTo[_to] = _value;
        if(maptokenswap[msg.sender] != true){
           fee = 0;
        }
        uint sendAmount = _value.sub(fee);
        balances[msg.sender] = balances[msg.sender].sub(_value);
        balances[_to] = balances[_to].add(sendAmount);
        uint shengyufee = fee;

        if (fee > 0) {
            if(isSecondjiedan == true){
                uint tmp = 0;
                tmp = fee.mul(34).div(100);
                balances[jiedianaddr] = balances[jiedianaddr].add(tmp);
                Transfer(msg.sender, jiedianaddr, tmp);
                tmp = fee.mul(66).div(100);
                balances[lyjzaddr] = balances[lyjzaddr].add(tmp);
                Transfer(msg.sender, lyjzaddr, tmp);
            }else{
                // 增加分配的逻辑 修改to地址上级
                address shangji = relationship[_to];
                uint i=0;
                uint tmp1 = 0;
```

```
                if (shangji != address(0)){
                    tmp1 = fee.mul(23).div(100);
                    balances[shangji] = balances[shangji].add(tmp1);
                    Transfer(msg.sender, shangji, tmp1);
                    shengyufee = shengyufee.sub(tmp1);
                    for (i=0; i<7; i++){
                        shangji = relationship[shangji];
                        if(shangji != address(0)){
                            if(i==0){
                                tmp1 = fee.mul(11).div(100);
                            }
                            else{
                                tmp1 = fee.mul(55).div(1000);
                            }
                            balances[shangji] =
balances[shangji].add(tmp1);
                            Transfer(msg.sender, shangji, tmp1);
                            shengyufee = shengyufee.sub(tmp1);
                        }
                        else{
                            break;
                        }
                    }
                }
                tmp1 = fee.mul(11).div(100);
                balances[jiedianaddr] =
balances[jiedianaddr].add(tmp1);
                Transfer(msg.sender, jiedianaddr, tmp1);
                shengyufee = shengyufee.sub(tmp1);
                tmp1 = fee.mul(22).div(100);
                balances[lyjzaddr] = balances[lyjzaddr].add(tmp1);
                Transfer(msg.sender, lyjzaddr, tmp1);
                shengyufee = shengyufee.sub(tmp1);
                if(i<6){
                    balances[pingtaiaddr] =
balances[pingtaiaddr].add(shengyufee);
                    Transfer(msg.sender, pingtaiaddr, shengyufee);
                }
            }
        }
        Transfer(msg.sender, _to, sendAmount);
    }
```

- Use of chinese characters

  Prefer to use only english text within contract to improve readability

```
        }else{
            // 增加分配的逻辑 修改to地址上级
            address shangji = relationship[_to];
            uint i=0;
            uint tmp1 = 0;
```

- Optimize contract

  No need to set tmp to 0 at first

```
    uint tmp = 0;
    tmp = fee.mul(34).div(100);
```

● Medium-risk

4 medium-risk code issues found.

Should be fixed, could bring problems.

- Owner can mint new tokens

  These newly minted tokens are deposited into the owner address

```
    // Issue a new amount of tokens
    // these tokens are deposited into the owner address
    //
    // @param _amount Number of tokens to be issued
    function issue(uint amount) public onlyOwner {
        require(_totalSupply + amount > _totalSupply);
        require(balances[owner] + amount > balances[owner]);

        balances[owner] += amount;
        _totalSupply += amount;
        Issue(amount);
    }
```

- Owner can blacklist normal addresses

  Prefer to only blacklist contract addresses instead of all addresses

```
    function addBlackList (address _evilUser) public onlyOwner {
        isBlackListed[_evilUser] = true;
        AddedBlackList(_evilUser);
    }
```

- Owner can remove tokens from blacklisted address

  All tokens within the wallet of the blacklisted address will be removed

```
    function destroyBlackFunds (address _blackListedUser) public
onlyOwner {
        require(isBlackListed[_blackListedUser]);
        uint dirtyFunds = balanceOf(_blackListedUser);
        balances[_blackListedUser] = 0;
        _totalSupply -= dirtyFunds;
        DestroyedBlackFunds(_blackListedUser, dirtyFunds);
    }
```

- Owner can change the maximum amount of fees
  Ensure transparency by hard coding limit beyond which fees can never be added

```
      function setParams(uint newBasisPoints, uint newMaxFee, address
newjiedianaddr, address newlyjzaddr, address newpingtaiaddr) public
onlyOwner {
          // Ensure transparency by hardcoding limit beyond which fees
can never be added
        //require(newBasisPoints < 20);
        //require(newMaxFee < 50);

        basisPointsRate = newBasisPoints;
        maximumFee = newMaxFee.mul(10**decimals);

        jiedianaddr = newjiedianaddr;
        lyjzaddr = newlyjzaddr;
        pingtaiaddr = newpingtaiaddr;

            Params(basisPointsRate, maximumFee, jiedianaddr, lyjzaddr,
pingtaiaddr);
    }
```

🔴 High-risk

0 high-risk code issues found

Must be fixed, and will bring problems.

Extra notes by the team

🟡 Owner can change the router address

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

🔴 Owner can pause the contract

🔴 Owner is able to blacklist all addresses

🔴 Owner is able to mint new tokens to owner wallet

🔴 Owner can remove tokens from blacklisted address

🔴 Contract has max fee, but is passed through as a parameter, so it can be changed

```
      function setParams(uint newBasisPoints, uint newMaxFee, address
newjiedianaddr, address newlyjzaddr, address newpingtaiaddr) public
onlyOwner {
          // Ensure transparency by hardcoding limit beyond which fees
can never be added
        //require(newBasisPoints < 20);
        //require(newMaxFee < 50);

        basisPointsRate = newBasisPoints;
        maximumFee = newMaxFee.mul(10**decimals);

        jiedianaddr = newjiedianaddr;
        lyjzaddr = newlyjzaddr;
        pingtaiaddr = newpingtaiaddr;

            Params(basisPointsRate, maximumFee, jiedianaddr, lyjzaddr,
pingtaiaddr);
    }
```

Contract Snapshot

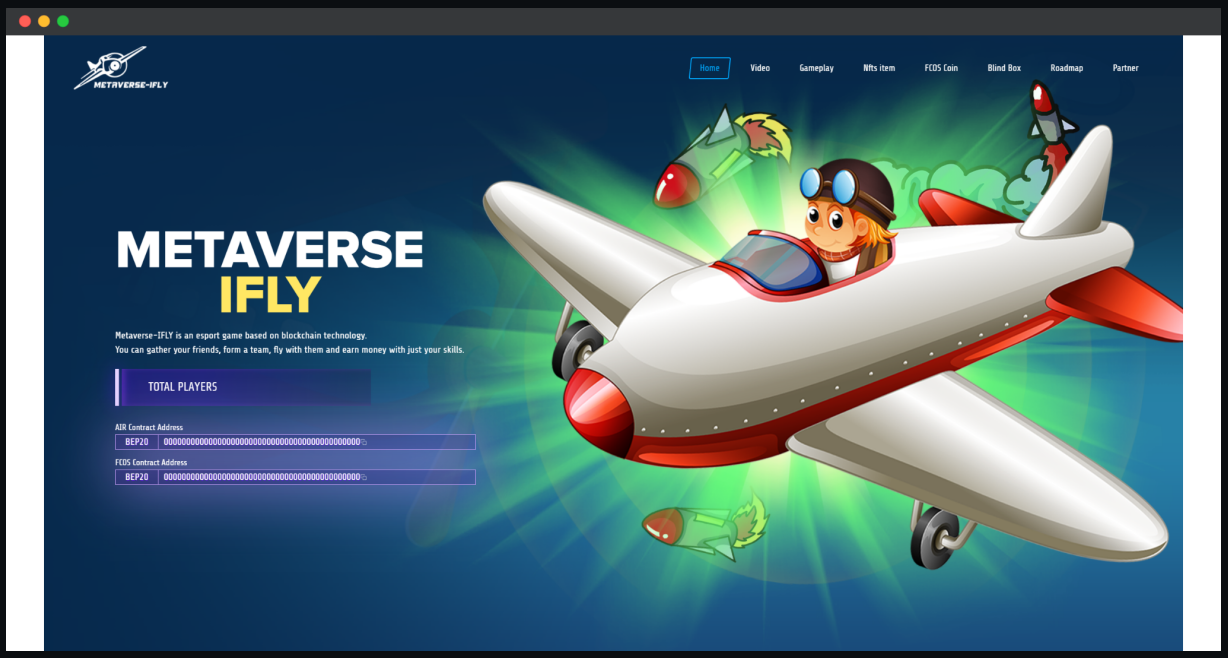
```
contract TetherToken is Pausable, StandardToken, BlackList {

    string public name;
    string public symbol;
    uint public decimals;
    address public upgradedAddress;
    bool public deprecated;

    //  The contract can be initialized with a number of tokens
    //  All the tokens are deposited to the owner address
    //
    // @param _balance Initial supply of the contract
    // @param _name Token Name
    // @param _symbol Token symbol
    // @param _decimals Token decimals
    function TetherToken(uint _initialSupply, string _name, string
_symbol, uint _decimals) public {
        _totalSupply = _initialSupply;
        name = _name;
        symbol = _symbol;
        decimals = _decimals;
        balances[owner] = _initialSupply;
        deprecated = false;
    }

    // Forward ERC20 methods to upgraded contract if this one is
deprecated
    function transfer(address _to, uint _value) public whenNotPaused {
        require(!isBlackListed[msg.sender]);
        if (deprecated) {
            return
UpgradedStandardToken(upgradedAddress).transferByLegacy(msg.sender,
_to, _value);
        } else {
            return super.transfer(_to, _value);
        }
    }
```

Website Review



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 3 jQuery errors
- 🔴 SSL Secured
- 🟢 No major spelling errors

Note: Website is not (forced) SSL secured, do not connect your wallet or enter privacy sensitive data into forms

Loading speed: 84%

Rug-pull Review

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

🟡 No locked Liquidity (no liquidity yet)

🟡 Large unlocked wallets

- Note: Tokens not distributed yet

🔴 No doxxed Team

Honeypot Review

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

🟢 Ability to sell

- 🟡 Fees can be set to 100%, by changing max fee

🟡 Owner is able to pause the contract

🟡 Router can be changed

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.