



# Coinsult

## Advanced Manual Smart Contract Audit



**Project:** Xuanyuan alliance games

**Website:** <https://www.xobgame.com/>

**Low-risk**

4 low-risk code  
issues found

**Medium-risk**

0 medium-risk code  
issues found

**High-risk**

0 high-risk code  
issues found

**Contract address**

0x5e7aBe4480d91a605a627a3F8c9007af3dD5b170

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	0xb6ee5ef7a70c5c78f811b4367ccc53b4bd8ee4ec	1,000,000,000	100.0000%

# Source code

Coinsult was commissioned by Xuanyuan alliance games to perform an audit based on the following smart contract:

<https://www.bscscan.com/address/0x5e7aBe4480d91a605a627a3F8c9007af3dD5b170#code>

# Manual Code Review

## ● Low-risk

4 low-risk code issues found.

Could be fixed, will not bring problems.

- Dangerous strict equalities

Don't use strict equality to determine if an account has enough Ether or tokens.

```
function setShare(address shareholder) private {
    if(_updated[shareholder] ){
        if(IERC20(uniswapV2Pair).balanceOf(shareholder) == 0)
quitShare(shareholder);
        return;
    }
    if(IERC20(uniswapV2Pair).balanceOf(shareholder) == 0)
return;

    addShareholder(shareholder);
    _updated[shareholder] = true;

}
```

- Block.timestamp can be manipulated by miners.  
Avoid relying on block.timestamp.

More information:

<https://github.com/crytic/slither/wiki/Detector-Documentation#block-timestamp>

```
// make the swap

uniswapV2Router.swapExactTokensForTokensSupportingFeeOnTransferTokens(
    tokenAmount,
    0,
    path,
    address(this),
    block.timestamp
);
```

- Literals with many digits are difficult to read and review.  
Recommendation: Use Ether suffix, Time suffix, or The scientific notation

```
uint256 distributorGas = 500000 ;
```

- Divide before multiply  
Solidity integer division might truncate. As a result, performing multiplication before division can sometimes avoid loss of precision.

```
        _tOwned[marketAddr] =  
_tOwned[marketAddr].add(tAmount.div(10000).mul(_inviterFee));  
        emit Transfer(sender, marketAddr,  
tAmount.div(10000).mul(_inviterFee));
```

### ● **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 exclude from fees

- Owner can blacklist

```
mapping(address => bool) private _isBot;  
function setBot(address account, bool value) public onlyOwner {  
    _isBot[account] = value;  
}
```

# Contract Snapshot

```
abstract contract Ownable {
    address private _owner;
    address private _previousOwner;
    uint256 private _lockTime;

    event OwnershipTransferred(address indexed previousOwner, address
indexed newOwner);

    constructor () {
        address msgSender = msg.sender;
        _owner = msgSender;
        emit OwnershipTransferred(address(0), msgSender);
    }

    function owner() public view returns (address) {
        return _owner;
    }

    modifier onlyOwner() {
        require(_owner == msg.sender, "Ownable: caller is not the
owner");
        _;
    }

    function renounceOwnership() public virtual onlyOwner {
        emit OwnershipTransferred(_owner, address(0));
        _owner = address(0);
    }

    function transferOwnership(address newOwner) public virtual
onlyOwner {
        require(newOwner != address(0), "Ownable: new owner is the zero
address");
        emit OwnershipTransferred(_owner, newOwner);
        _owner = newOwner;
    }
}
```

# Website Review

Website not available

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: ? %



# 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 - Tokens not yet distributed
- No doxxed Team

# Honeypot Review

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

- Ability to sell
- Owner is not able to pause trading
- Router 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.