Coinsult

Advanced Manual Smart Contract Audit

Project: SHIBAMOVE

Website: https://shibmove.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

0x41Aa00be6566E00EE5Dc2C2D48A6A437e833e120

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	500,000,000,000,000	50.0000%
2	0x146321fdfbb7257e4ee08db7fbd4b8b327852ab9	390,330,000,000,000	39.0330%
3	0x84dafb4a6d9e01c2ae9ada91eed6bae87b654961	60,000,000,000,000	6.0000%
4	0x9bc1518951c30c7c643a6371f1add230fb422ff6	29,420,000,000,000	2.9420%
5	0x93ff847dfe83c685c1a284d37112e1d3067e1000	6,000,000,000,000	0.6000%

Source Code

Coinsult was comissioned by SHIBAMOVE to perform an audit based on the following smart contract:

https://bscscan.com/address/0x41Aa00be6566E00EE5Dc2C2D48A6A437e833e120#code

Manual Code Review

In this audit report we will highlight all these issues:

🟢 Low-Risk

5 low-risk code issues found

🟡 Medium-Risk

0 medium-risk code issues found

🔴 High-Risk

0 high-risk code issues found

The detailed report continues on the next page...

● **Low-Risk:** Could be fixed, will not bring problems.

Avoid relying on block.timestamp

block.timestamp can be manipulated by miners.

```
lastClaimTime = lastClaimTimes[account];

nextClaimTime = lastClaimTime &gt; 0 ?
                            lastClaimTime.add(claimWait) :
                            0;

secondsUntilAutoClaimAvailable = nextClaimTime &gt; block.timestamp ?
                                            nextClaimTime.sub(block.timestamp) :
                                            0;
```

Recommendation

Do not use `block.timestamp`, now or `blockhash` as a source of randomness

Exploit scenario

```
contract Game {

    uint reward_determining_number;

    function guessing() external{
      reward_determining_number = uint256(block.blockhash(10000)) % 10;
    }
}
```

Eve is a miner. Eve calls `guessing` and re-orders the block containing the transaction. As a result, Eve wins the game.

● **Low-Risk:** Could be fixed, will not bring problems.

Too many digits

Literals with many digits are difficult to read and review.

```
function updateGasForProcessing(uint256 newValue) public onlyOwner {
    require(newValue &gt;= 200000 &amp;&amp; newValue &lt;= 500000, &quot;GasForProcessing must be b
    require(newValue != gasForProcessing, &quot;Cannot update gasForProcessing to same value&quot;);
    emit GasForProcessingUpdated(newValue, gasForProcessing);
    gasForProcessing = newValue;
}
```

Recommendation

Use: Ether suffix, Time suffix, or The scientific notation

Exploit scenario

```
contract MyContract{
    uint 1_ether = 10000000000000000000;
}
```

While `1_ether` looks like `1 ether`, it is `10 ether`. As a result, it’s likely to be used incorrectly.

● **Low-Risk:** Could be fixed, will not bring problems.

No zero address validation for some functions

Detect missing zero address validation.

```
function setMarketingWallet(address payable wallet) external onlyOwner{
    _marketingWalletAddress = wallet;
}
```

Recommendation

Check that the new address is not zero.

Exploit scenario

```
contract C {

  modifier onlyAdmin {
    if (msg.sender != owner) throw;
    _;
  }

  function updateOwner(address newOwner) onlyAdmin external {
    owner = newOwner;
  }
}
```

Bob calls `updateOwner` without specifying the `newOwner`, soBob loses ownership of the contract.

● **Low-Risk:** Could be fixed, will not bring problems.

Unchecked transfer

The return value of an external transfer/transferFrom call is not checked.

```
function swapAndSendToFee(uint256 tokens) private  {
    uint256 initialCAKEBalance = IERC20(rewardToken).balanceOf(address(this));
    swapTokensForCake(tokens);
    uint256 newBalance = (IERC20(rewardToken).balanceOf(address(this))).sub(initialCAKEBalance);
    IERC20(rewardToken).transfer(_marketingWalletAddress, newBalance);
    AmountMarketingFee = AmountMarketingFee - tokens;
}
```

Recommendation

Use `SafeERC20`, or ensure that the transfer/transferFrom return value is checked.

Exploit scenario

```
contract Token {
    function transferFrom(address _from, address _to, uint256 _value) public returns (bool success);
}
contract MyBank{
    mapping(address => uint) balances;
    Token token;
    function deposit(uint amount) public{
        token.transferFrom(msg.sender, address(this), amount);
        balances[msg.sender] += amount;
    }
}
```

Several tokens do not revert in case of failure and return false. If one of these tokens is used in `MyBank`, `deposit` will not revert if the transfer fails, and an attacker can call `deposit` for free..

● **Low-Risk:** Could be fixed, will not bring problems.

Missing events arithmetic

Detect missing events for critical arithmetic parameters.

```
function setBuyTaxes(uint256 liquidity, uint256 rewardsFee, uint256 marketingFee, uint256 deadFee) e
    require(rewardsFee.add(liquidity).add(marketingFee).add(deadFee) &lt;= 25, &quot;Total buy fee i
    buyTokenRewardsFee = rewardsFee;
    buyLiquidityFee = liquidity;
    buyMarketingFee = marketingFee;
    buyDeadFee = deadFee;

}
```

Recommendation

Emit an event for critical parameter changes.

Exploit scenario

```
contract C {

  modifier onlyAdmin {
    if (msg.sender != owner) throw;
    _;
  }

  function updateOwner(address newOwner) onlyAdmin external {
    owner = newOwner;
  }
}
```

`updateOwner()` has no event, so it is difficult to track off-chain changes in the buy price.

Owner privileges

- 🟢 Owner cannot set fees higher than 25%
- 🟢 Owner cannot pause trading
- 🟢 Owner cannot change max transaction amount
- 🟡 Owner can exclude from fees
- 🔴 Owner can blacklist addresses

Extra notes by the team

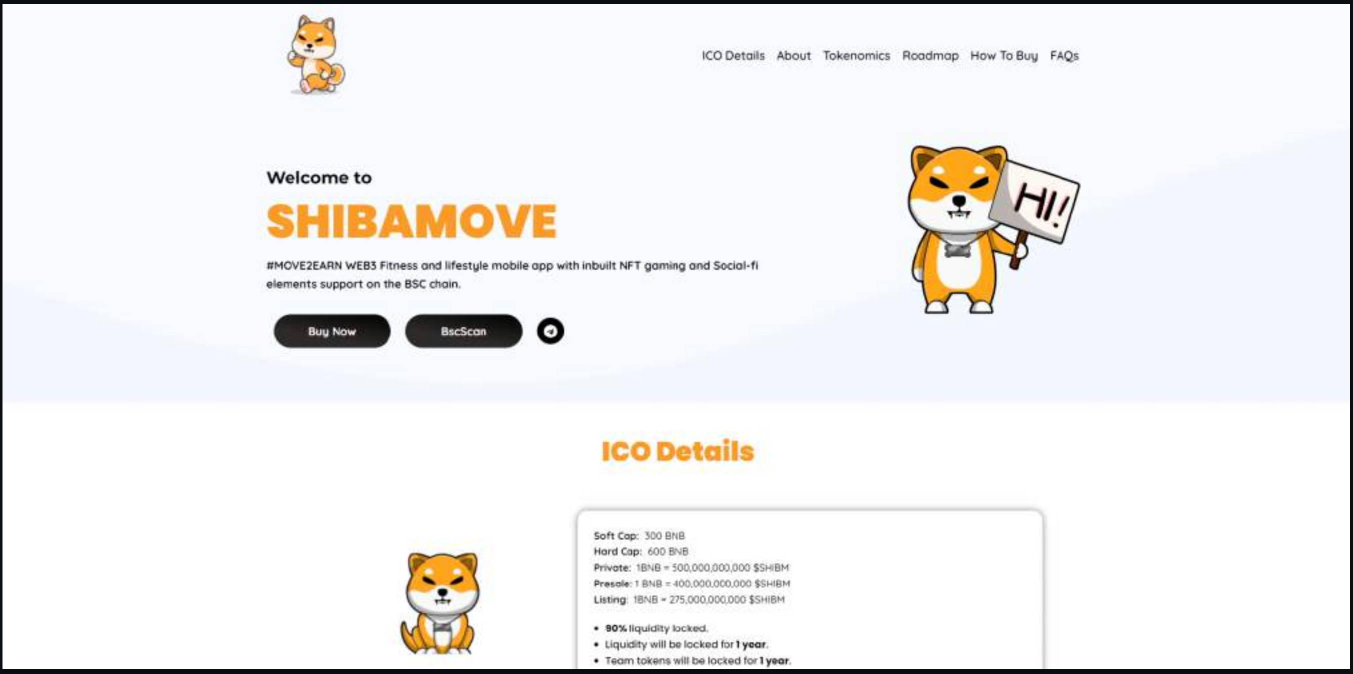
No notes

Contract Snapshot

```
contract CoinToken is ERC20, Ownable {
using SafeMath for uint256;

IUniswapV2Router02 public uniswapV2Router;
address public  uniswapV2Pair;

bool private swapping;

TokenDividendTracker public dividendTracker;

address public rewardToken;

uint256 public swapTokensAtAmount;

uint256 public buyTokenRewardsFee;
uint256 public sellTokenRewardsFee;
uint256 public buyLiquidityFee;
uint256 public sellLiquidityFee;
uint256 public buyMarketingFee;
uint256 public sellMarketingFee;
uint256 public buyDeadFee;
uint256 public sellDeadFee;
uint256 public AmountLiquidityFee;
uint256 public AmountTokenRewardsFee;
uint256 public AmountMarketingFee;

address public _marketingWalletAddress;
address private _node;

address public deadWallet = 0x000000000000000000000000000000000000dEaD;
mapping(address =&gt; bool) public _isEnemy;
```

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.



Welcome to

SHIBAMOVE

#MOVE2EARN WEB3 Fitness and lifestyle mobile app with inbuilt NFT gaming and Social-fi elements support on the BSC chain.

ICO Details

Soft Cap: 300 BNB
Hard Cap: 600 BNB
Private: 1BNB = 500,000,000,000 $SHIBM
Presale: 1 BNB = 400,000,000,000 $SHIBM
Listing: 1BNB = 275,000,000,000 $SHIBM

- **90%** liquidity locked.
- Liquidity will be locked for **1 year**.
- Team tokens will be locked for **1 year**.

● Mobile Friendly

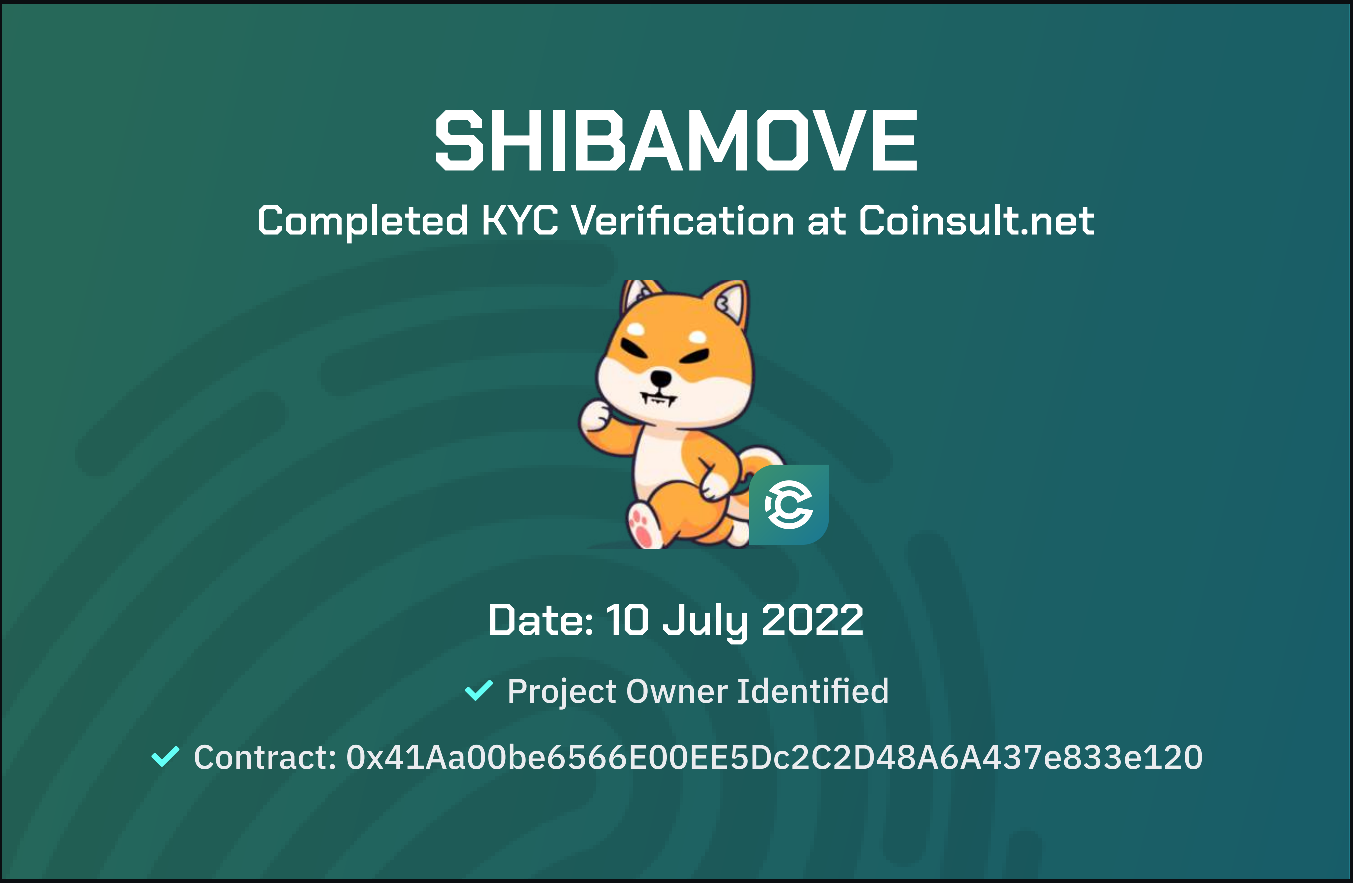
● Does not contain jQuery errors

● SSL Secured

● No major spelling errors

Project Overview

● KYC verified by Coinsult



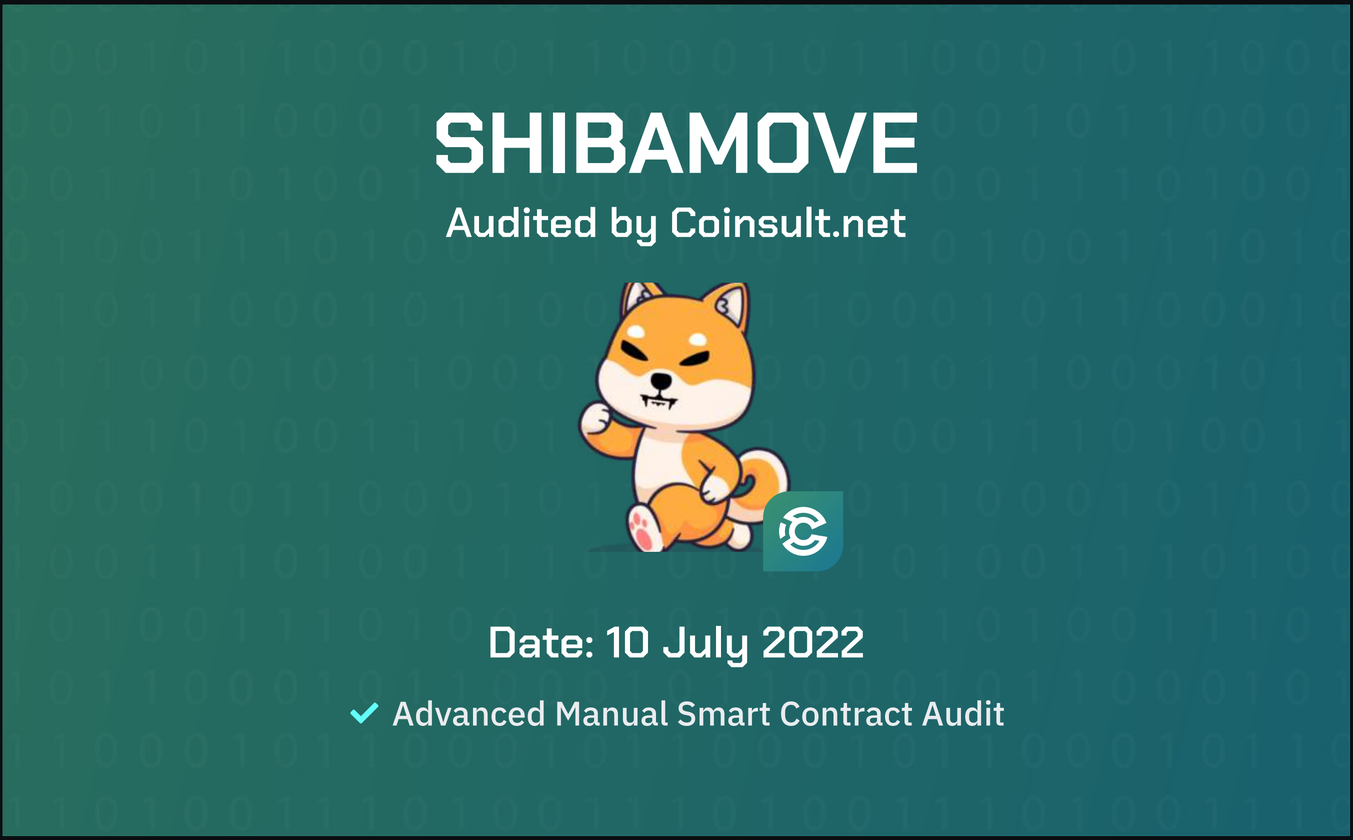
SHIBAMOVE

Completed KYC Verification at Coinsult.net

Date: 10 July 2022

✓ Project Owner Identified

✓ Contract: 0x41Aa00be6566E00EE5Dc2C2D48A6A437e833e120



SHIBAMOVE

Audited by Coinsult.net

Date: 10 July 2022

✓ Advanced Manual Smart Contract Audit