Coinsult

Advanced Manual Smart Contract Audit

Project: Metacats
Website: https://metacatstoken.com/

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

Contract address
0xc3e39c04f6b638e3a4eb3b17b97014907c8553b0

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

Total Supply:	1,000,000,000,000,000
Total Holders:	1

Top 10 holders:

Rank	Address	Quantity (Token)	Percentage
1	0x86af3d9d128ca5753e3ed50de376e03395e4c3ba	1,000,000,000,000,000	100.0000%

The top 100 holders collectively own 100.00% (1,000,000,000,000,000 Tokens) of Metacats

Note: This is a snapshot of when the audit was performed.

Source code

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

https://bscscan.com/address/0xc3e39c04f6b638e3a4eb3b17b97014907c8553b0#code

Manual Code Review

● Low-risk

4 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 sender, address recipient, uint256
amount) private returns (bool) {

        require(sender != address(0), "ERC20: transfer from the zero
address");
        require(recipient != address(0), "ERC20: transfer to the zero
address");
        require(!_isbclisted[recipient] && !_isbclisted[sender],
"bclisted address");

        if(inSwapAndLiquify)
        {
            return _basicTransfer(sender, recipient, amount);
        }
        else
        {
            if (!isExcludedFromFee[sender] &&
!isExcludedFromFee[recipient] && _sellReserveFee > 0) {
                uint _sellReserveFeeAmount =
.................
```

- Function which sends eth to arbitrary destination
  Ensure that an arbitrary user cannot withdraw unauthorized funds. More information: Slither

```
    function addLiquidity(uint256 tokenAmount, uint256 ethAmount)
private {
        // approve token transfer to cover all possible scenarios
        _approve(address(this), address(uniswapV2Router), tokenAmount);

        // add the liquidity
        uniswapV2Router.addLiquidityETH{value: ethAmount}(
            address(this),
            tokenAmount,
            0, // slippage is unavoidable
            0, // slippage is unavoidable
            deadAddress,
            block.timestamp
        );
    }
```

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

```
    function getTime() public view returns (uint256) {
        return block.timestamp;
    }

    function lock(uint256 time) public virtual onlyOwner {
        _previousOwner = _owner;
        _owner = address(0);
        _lockTime = block.timestamp + time;
        emit OwnershipTransferred(_owner, address(0));
    }

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

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

```
            if (!isExcludedFromFee[sender] &&
!isExcludedFromFee[recipient] && _sellReserveFee > 0) {
                uint _sellReserveFeeAmount =
amount.div(100).mul(_sellReserveFee);
                amount = amount.sub(_sellReserveFeeAmount);
            }
```

● 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

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

- **Owner able to change router.**

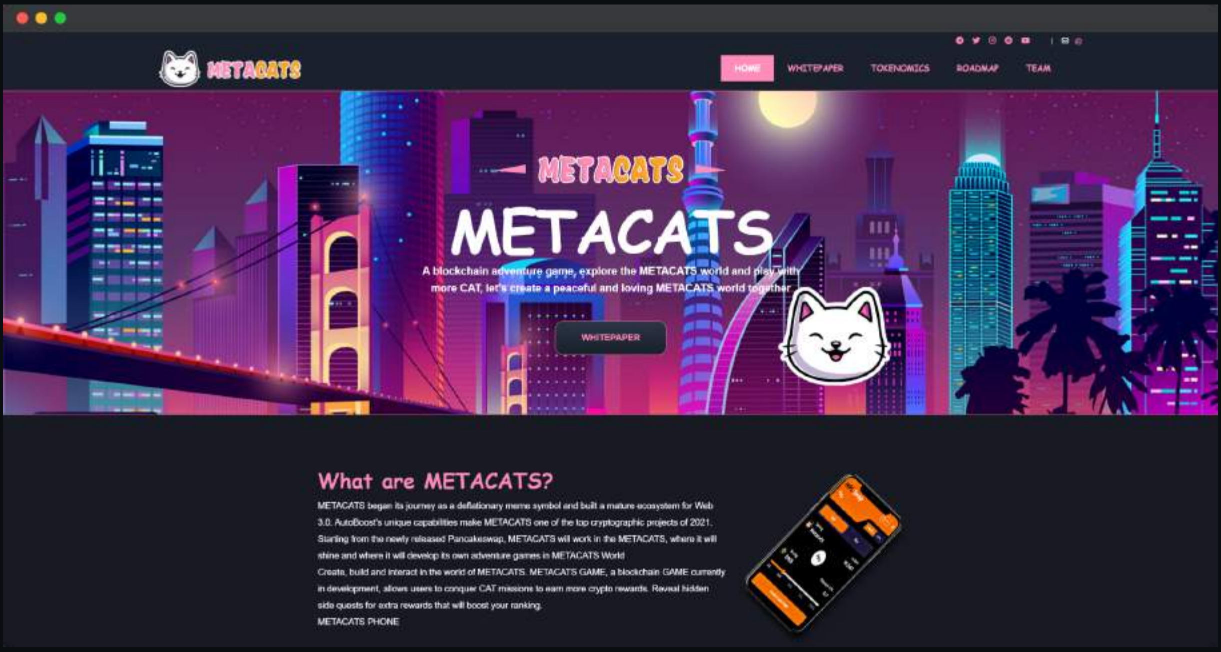
- The ownership of the contract isn’t renounced

- Owner can whitelist addresses from fees.

Contract Snapshot

```
contract MetaCats is Context, IERC20, Ownable {

    using SafeMath for uint256;
    using Address for address;

    string private _name = "MetaCats";
    string private _symbol = "MetaCats";
    uint8 private _decimals = 9;

    uint public launchedBlock;
    uint public killblock = 2;
    bool public isLaunch = false;
    address payable public marketingWalletAddress =
payable(0x8430d70ca4Ffeaaa202bC67088376B02Ffd0B05c); // Marketing Address
    address payable public teamWalletAddress =
payable(0xAeB466cd41B280794d54f91d99eC169f74B80469); // Team Address
    address public immutable deadAddress =
0x000000000000000000000000000000000000dEaD;

    mapping (address => uint256) _balances;
    mapping (address => mapping (address => uint256)) private _allowances;

    mapping (address => bool) public isExcludedFromFee;
    mapping (address => bool) public isWalletLimitExempt;
    mapping (address => bool) public isTxLimitExempt;
    mapping (address => bool) public isMarketPair;
    mapping (address => bool) private _isbclisted;

    uint256 public _buyLiquidityFee = 2;
    uint256 public _buyMarketingFee = 4;
    uint256 public _buyTeamFee = 4;

    uint256 public _sellLiquidityFee = 2;
    uint256 public _sellMarketingFee = 4;
    uint256 public _sellTeamFee = 4;

    uint256 public _sellReserveFee = 1;

    uint256 public _liquidityShare = 4;
    uint256 public _marketingShare = 10;
    uint256 public _teamShare = 10;

    uint256 public _totalTaxIfBuying = 12;
    uint256 public _totalTaxIfSelling = 12;
```

Website Review



METACATS

A blockchain adventure game, explore the METACATS world and play with more CAT, let's create a peaceful and loving METACATS world together

What are METACATS?

METACATS began its journey as a deflationary meme symbol and built a mature ecosystem for Web 3.0. AutoBoost's unique capabilities make METACATS one of the top cryptographic projects of 2021. Starting from the newly released Pancakeswap, METACATS will work in the METACATS, where it will shine and where it will develop its own adventure games in METACATS World

Create, build and interact in the world of METACATS. METACATS GAME, a blockchain GAME currently in development, allows users to conquer CAT missions to earn more crypto rewards. Reveal hidden side quests for extra rewards that will boost your ranking.

METACATS PHONE

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

Note: Website not fully functional yet, multiple buttons do not work properly yet.

Loading speed: 86%

Rug-pull Review

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

🟡 Locked Liquidity (no liquidity pool yet)

🟡 Large unlocked wallets (100%)

Note: During contract creation the owner wallet will receive 100% of the supply. The dev team will probably divide this supply to other addresses later in time.

🔴 No doxxed Team

Honeypot Review

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

🟡 Ability to sell

Note: Owner can change selling fees up to 100% to prevent selling

🟢 Owner not able to prevent selling

🟡 Accurate liquidity pair

Note: Owner can change liquidity router

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.