



Coinsult

Advanced Manual Smart Contract Audit



Project: Zombinomy

Website: <https://zombinomy.net/>

Low-risk

3 low-risk code
issues found

Medium-risk

0 medium-risk code
issues found

High-risk

1 high-risk code
issues found

1 Acknowledged

Contract address

0x8881B26fd19Cff3aE89a62332eA241c3EE32FAAf

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

Total Supply: 100,000,000

Total Holders: 2

Top 10 holders:

Rank	Address	Quantity (Token)	Percentage
1	0xb44254d4c788257dd249927520cd697c7e3be04e	96,299,947.08	96.2999%
2	0xb751cb9185fb15216913f139e3bbf7cc97834dbb	3,700,052.92	3.7001%

The top 100 holders collectively own 100.00% (100,000,000.00 Tokens) of Zombinomy

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

Source code

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

<https://bscscan.com/address/0x8881B26fd19Cff3aE89a62332eA241c3EE32FAAf#code>

Manual Code Review

● Low-risk

3 low-risk code issues found.

Could be fixed, will not bring problems.

- Owner entitled to change the transaction fees

```
function setMarketingFee(uint256 _marketingFee) external onlyOwner
{
    marketingFee = _marketingFee;
}
```

- Owner entitled to whitelist from fees

```
function excludeFromFees(address account, bool excluded) public
onlyOwner {
    require(!_isExcludedFromFees[account] != excluded, "MicroPets:
Account is already the value of 'excluded'");
    _isExcludedFromFees[account] = excluded;

    emit ExcludeFromFees(account, excluded);
}
```

- The ownership of the contract isn't renounced

● Medium-risk

0 medium-risk code issues found.

Should be fixed, could bring problems.

● High-risk

1 high-risk code issues found | ✓ 1 Acknowledged

Must be fixed, and will bring problems.

- Owner can mint tokens

```
function _mint(address account, uint256 amount) internal virtual {  
    require(account != address(0), "ERC20: mint to the zero  
address");  
  
    _beforeTokenTransfer(address(0), account, amount);  
  
    _totalSupply = _totalSupply.add(amount);  
    _balances[account] = _balances[account].add(amount);  
    emit Transfer(address(0), account, amount);  
}
```

Acknowledged by the Zombinomy team:

“We need that mint function for a smart contract deployed on top of our contract which enables payments with our coin in our metaverse”

Extra notes by the team

Comments by the team about the code in general.

Note 1:

setMarketingFee(uint256) (#1017-1019) should emit an event for:

- marketingFee = _marketingFee (#1018)

Emit an event for critical parameter changes.

Note 2:

ERC20._burn(address,uint256) (#887-895) is never used and should be removed

SafeMath.mod(uint256,uint256) (#541-543) is never used and should be removed

SafeMath.mod(uint256,uint256,string) (#557-560) is never used and should be removed

SafeMathInt.abs(int256) (#641-644) is never used and should be removed

SafeMathInt.add(int256,int256) (#632-636) is never used and should be removed

SafeMathInt.div(int256,int256) (#612-618) is never used and should be removed

SafeMathInt.mul(int256,int256) (#600-607) is never used and should be removed

SafeMathInt.sub(int256,int256) (#623-627) is never used and should be removed

SafeMathInt.toUint256Safe(int256) (#647-650) is never used and should be removed

SafeMathUint.toInt256Safe(uint256) (#658-662) is never used and should be removed

Note 3:

constructor() (#956-967) uses literals with too many digits:

- _mint(owner(),100000000 * (10 ** 18)) (#966)

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

Contract Snapshot

```
contract Zombinomy is ERC20, Ownable {

    using SafeMath for uint256;

    IUniswapV2Router02 public uniswapV2Router;
    address public uniswapV2Pair;
    uint256 public marketingFee = 1;
    // exclude from fees and max transaction amount
    mapping (address => bool) private _isExcludedFromFees;

    event ExcludeFromFees(address indexed account, bool isExcluded);
    event SetAutomatedMarketMakerPair(address indexed pair, bool
indexed value);

    constructor() public ERC20("Zombinomy", "ZOMB") {

        IUniswapV2Router02 _uniswapV2Router =
IUniswapV2Router02(0x10ED43C718714eb63d5aA57B78B54704E256024E);
        uniswapV2Pair =
IUniswapV2Factory(_uniswapV2Router.factory()).createPair(address(this),
_uniswapV2Router.WETH());

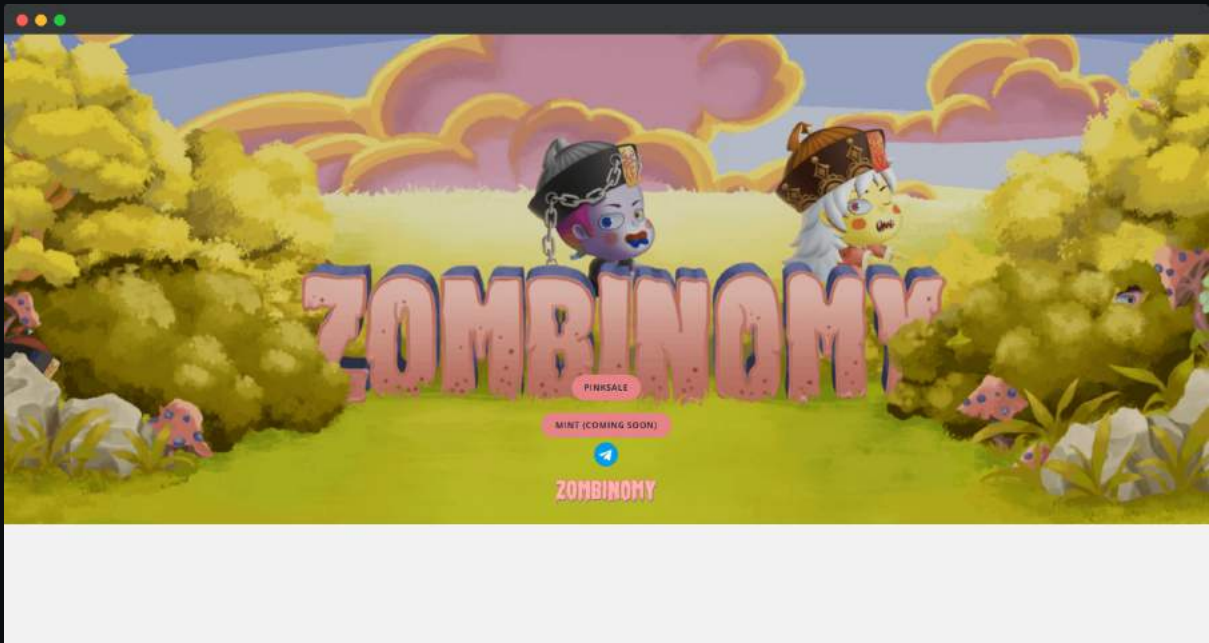
        uniswapV2Router = _uniswapV2Router;

        excludeFromFees(owner(), true);
        excludeFromFees(address(this), true);

        _mint(owner(), 1000000000 * (10**18));
    }

    receive() external payable {
```

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 (However some scrolling issues might occur)
- No jQuery errors
- SSL Secured
- Appropriate spelling

Loading speed: 92%

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 ⚠️
(at the moment of the audit)
- 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 higher than 25% by owner)
- Owner unable to prevent selling ✅
(But, fees can be set higher than 25% by owner)
- Accurate liquidity pair ✅

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.