



Coinsult

Advanced Manual Smart Contract Audit



Project: Metanftgenerator

Website: <https://metanftgenerator.site/>

Low-Risk

2 low-risk code
issues found

Medium-Risk

0 medium-risk code
issues found

High-Risk

0 high-risk code
issues found

Contract Address

0x08367609be03DaF26414b2dD86F8F696eE205064

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	0xf38d55fcfc692a9d7b940f42fa0c1293400eb378	1,000,000,000	100.0000%

Source Code

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

<https://bscscan.com/address/0x08367609be03daf26414b2dd86f8f696ee205064#code>

Manual Code Review

In this audit report we will highlight all these issues:

Low-Risk

2 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...

To

Lit

```
_totalSupply = 1000000000000000000000000000
```

Re

Us

Ex

```
contract MyContract{
    uint 1_ether = 1000000000000000000;
}
```

W

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

Redundant Statements

Detect the usage of redundant statements that have no effect.

```
function _msgData() internal view returns (bytes memory) {
    this; // silence state mutability warning without generating bytecode - see https://github.com/ethereum/solidity/issues/2318
    return msg.data;
}
```

Recommendation

Remove redundant statements if they congest code but offer no value.

Exploit scenario

```
contract RedundantStatementsContract {

    constructor() public {
        uint; // Elementary Type Name
        bool; // Elementary Type Name
        RedundantStatementsContract; // Identifier
    }

    function test() public returns (uint) {
        uint; // Elementary Type Name
        assert; // Identifier
        test; // Identifier
        return 777;
    }
}
```

Each commented line references types/identifiers, but performs no action with them, so no code will be generated for such statements and they can be removed.

Owner privileges

- Owner cannot pause trading
- Owner cannot change max transaction amount
- Owner can set fees higher than 25%

Extra notes by the team

No notes

Contract Snapshot

```
contract BEP20Token is Context, IBEP20, Ownable {
    using SafeMath for uint256;

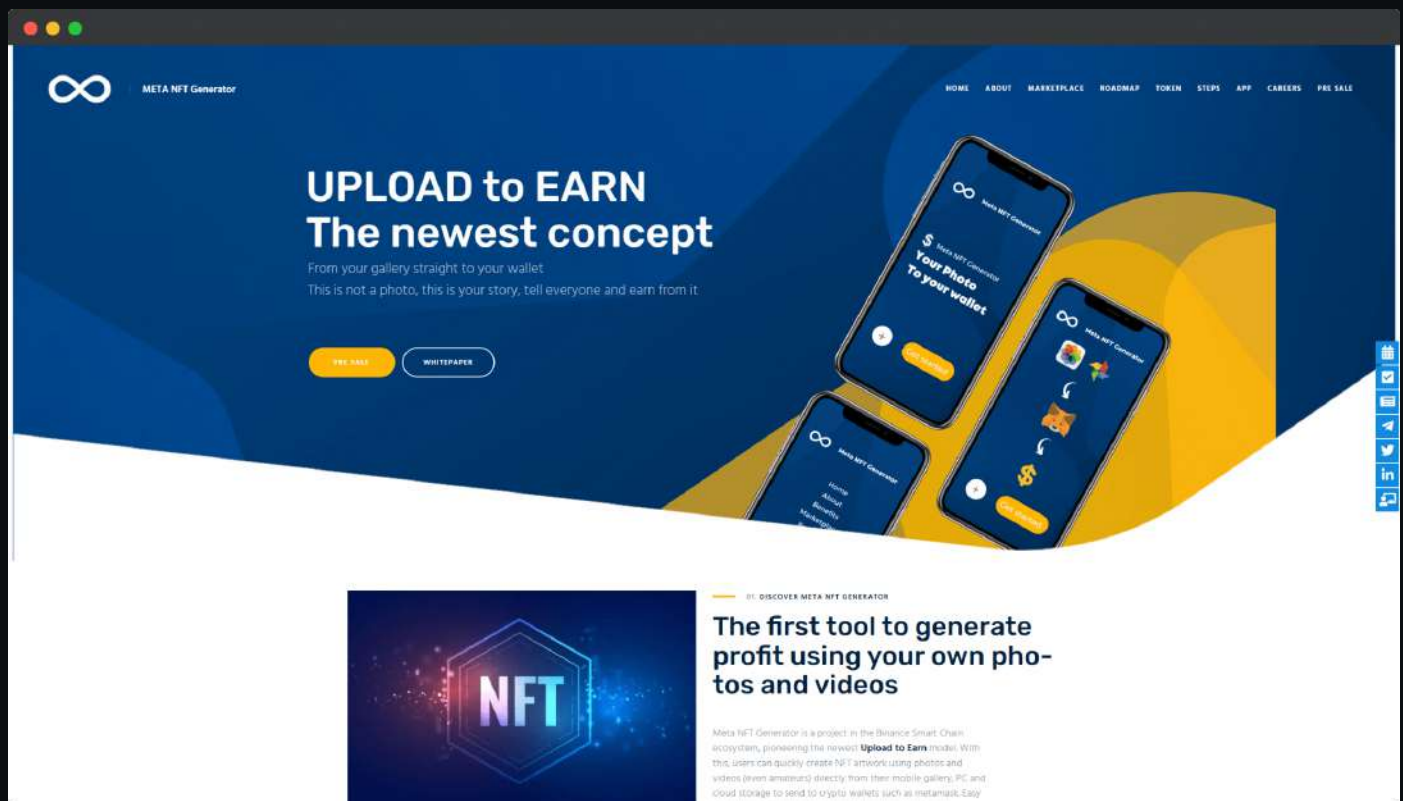
    mapping (address => uint256) private _balances;

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

    uint256 private _totalSupply;
    uint8 private _decimals;
    string private _symbol;
    string private _name;
```


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
- Does not contain jQuery errors
- SSL Secured
- No major spelling errors

Project Overview

● Not KYC verified by Coinsult

AUDITED
BY COINSULT.NET

