



# Smart Contract Security Audit Report

[2021]



The SlowMist Security Team received the Coin98 team's application for smart contract security audit of the C98MultiSend on 2021.08.12. The following are the details and results of this smart contract security audit:

**Token Name :**

C98MultiSend

**The contract address :**

<https://github.com/coin98/coin98-multisend>

commit: 664854045a64df872cdad3eb5e0d37202a2f4c6b

<https://etherscan.io/address/0xaec945e04baf28b135fa7c640f624f8d90f1c3a6>

<https://bscscan.com/address/0x2E1D30460265bFEBedacf5bb6f9A80F0E74B7498>

**The audit items and results :**

(Other unknown security vulnerabilities are not included in the audit responsibility scope)

NO.	Audit Items	Result
1	Replay Vulnerability	Passed
2	Denial of Service Vulnerability	Passed
3	Race Conditions Vulnerability	Passed
4	Authority Control Vulnerability	Passed
5	Integer Overflow and Underflow Vulnerability	Passed
6	Gas Optimization Audit	Passed
7	Design Logic Audit	Passed
8	Uninitialized Storage Pointers Vulnerability	Passed
9	Arithmetic Accuracy Deviation Vulnerability	Passed

NO.	Audit Items	Result
10	"False top-up" Vulnerability	Passed
11	Malicious Event Log Audit	Passed
12	Scoping and Declarations Audit	Passed
13	Safety Design Audit	Passed

**Audit Result :** Passed

**Audit Number :** 0x002108130001

**Audit Date :** 2021.08.12 - 2021.08.13

**Audit Team :** SlowMist Security Team

**Summary conclusion :** This is a contract that contains the MultiSend section. And the contract does not have the Overflow and the Race Conditions issue.

### The source code:

```
//SlowMist// The contract does not have the Overflow and the Race Conditions issue
pragma solidity ^0.4.23;

/**
 * @title ERC20 interface
 * @dev see https://github.com/ethereum/EIPs/issues/20
 */
contract ERC20 {
    uint public _totalSupply;
    function totalSupply() public view returns (uint);
    function balanceOf(address who) public view returns (uint);
    function transfer(address to, uint value) public;
    function allowance(address owner, address spender) public view returns (uint);
    function transferFrom(address from, address to, uint value) public;
    function approve(address spender, uint value) public;
}

contract C98MultiSend {
    address eth_address = 0xC02aaA39b223FE8D0A0e5C4F27eAD9083C756Cc2;
```

```

event transfer(address from, address to, uint amount,address tokenAddress);

// Transfer multi main network coin
// Example ETH, BSC, HT
function transferMulti(address[] receivers, uint256[] amounts) public payable {
    require(msg.value != 0 && msg.value == getTotalSendingAmount(amounts));
    for (uint256 i = 0; i < amounts.length; i++) {
        receivers[i].transfer(amounts[i]);
        emit transfer(msg.sender, receivers[i], amounts[i], eth_address);
    }
}

// Transfer multi token ERC20
function transferMultiToken(address tokenAddress, address[] receivers, uint256[]
amounts) public {
    require(receivers.length == amounts.length && receivers.length != 0);
    ERC20 token = ERC20(tokenAddress);

    for (uint i = 0; i < receivers.length; i++) {
        require(amounts[i] > 0 && receivers[i] != 0x0);
        token.transferFrom(msg.sender,receivers[i], amounts[i]);

        emit transfer(msg.sender, receivers[i], amounts[i], tokenAddress);
    }
}

function getTotalSendingAmount(uint256[] _amounts) private pure returns (uint
totalSendingAmount) {
    for (uint i = 0; i < _amounts.length; i++) {
        require(_amounts[i] > 0);
        totalSendingAmount += _amounts[i];
    }
}
}

```

## Statement

SlowMist issues this report with reference to the facts that have occurred or existed before the issuance of this report, and only assumes corresponding responsibility based on these.

For the facts that occurred or existed after the issuance, SlowMist is not able to judge the security status of this project, and is not responsible for them. The security audit analysis and other contents of this report are based on the documents and materials provided to SlowMist by the information provider till the date of the insurance report (referred to as "provided information"). SlowMist assumes: The information provided is not missing, tampered with, deleted or concealed. If the information provided is missing, tampered with, deleted, concealed, or inconsistent with the actual situation, the SlowMist shall not be liable for any loss or adverse effect resulting therefrom. SlowMist only conducts the agreed security audit on the security situation of the project and issues this report. SlowMist is not responsible for the background and other conditions of the project.



**Official Website**  
[www.slowmist.com](http://www.slowmist.com)



**E-mail**  
[team@slowmist.com](mailto:team@slowmist.com)



**Twitter**  
[@SlowMist\\_Team](https://twitter.com/SlowMist_Team)



**Github**  
<https://github.com/slowmist>