题目二

pragma solidity ^0.4.18;

contract Token2{

mapping (address => uint) allocations;

address private \_owner;

/\* constructor \*/

function consturctor() public payable {

\_owner = msg.sender;

allocations[\_owner] = msg.value;

}

function allocate() public payable {

allocations[msg.sender] += msg.value;

}

function sendAllocation(address allocator) public {

require(allocations[allocator] > 0);

allocator.transfer(allocations[allocator]);

}

function owner() public view returns(address) {

return \_owner;

}

modifier onlyOwner() {

require(isOwner());

\_;

}

function isOwner() public view returns(bool) {

return msg.sender == \_owner;

}

function collectAllocations() public onlyOwner {

msg.sender.transfer(this.balance);

}

function allocatorBalance(address allocator) public view returns (uint) {

return allocations[allocator];

}

}

题目说明：

找出代码中的漏洞，提交合约调用方式，获取合约所有权。